Open Source Used In DNAC Platform

Arica

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Text Part Number: 78EE117C99-1296472214
This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number 78EE117C99-1296472214

Contents

1.1 simple-odata-server 1.1.1
  1.1.1 Available under license
1.2 script-manager 0.10.2
  0.10.2 Available under license
1.3 isomorphic-fetch 2.2.1
  2.2.1 Available under license
1.4 ansi-escapes 3.2.0
  3.2.0 Available under license
1.5 make-dir 3.1.0
  3.1.0 Available under license
1.6 cookie-signature 1.0.6
  1.0.6 Available under license
1.7 jetty-xml 9.4.34.v20201102
  9.4.34.v20201102 Available under license
1.8 babel-helpers 6.24.1
  6.24.1 Available under license
1.9 zlib 1.2.3
  1.2.3 Available under license
1.10 kerby-pkix 1.0.0
  1.0.0 Available under license
1.11 commondir 1.0.1
  1.0.1 Available under license
1.12 kind-of 6.0.3
  6.0.3 Available under license
1.13 to-object-path 0.3.0
  0.3.0 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Package</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.14</td>
<td>lodash.isequal</td>
<td>4.5.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.15</td>
<td>batik-svg-dom</td>
<td>1.13</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.16</td>
<td>metrics-jetty</td>
<td>4.1.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.17</td>
<td>groovy-test</td>
<td>3.0.8</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.18</td>
<td>commons-io</td>
<td>2.5</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.19</td>
<td>form-data</td>
<td>1.0.1</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0.1</td>
<td>license</td>
</tr>
<tr>
<td>1.20</td>
<td>@babel/highlight</td>
<td>7.16.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.21</td>
<td>promise</td>
<td>2.1.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.1.0</td>
<td>license</td>
</tr>
<tr>
<td>1.22</td>
<td>drill-format-ltsv</td>
<td>1.19.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.23</td>
<td>curator-x-discovery</td>
<td>5.1.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.24</td>
<td>raw-body</td>
<td>2.3.2</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.3.2</td>
<td>license</td>
</tr>
<tr>
<td>1.25</td>
<td>libpng</td>
<td>1.2.54</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ubuntu1.1</td>
<td></td>
</tr>
<tr>
<td>1.26</td>
<td>jol-core</td>
<td>0.9</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.27</td>
<td>randomatic</td>
<td>3.1.1</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.28</td>
<td>babel-plugin-syntax-async-functions</td>
<td>6.13.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.29</td>
<td>pinkie-promise</td>
<td>1.0.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0.0</td>
<td>license</td>
</tr>
<tr>
<td>1.30</td>
<td>property-expr</td>
<td>2.0.4</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.0.4</td>
<td>license</td>
</tr>
<tr>
<td>1.31</td>
<td>websocket-server</td>
<td>9.3.27.v20190418</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.32</td>
<td>babel-plugin-transform-es2015-spread</td>
<td>6.22.0</td>
<td>6.22.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td>1.33</td>
<td>wrap-ansi</td>
<td>7.0.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>1.34</td>
<td>listencount</td>
<td>1.0.1</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0.1</td>
<td>license</td>
</tr>
</tbody>
</table>
1.34.1 Available under license
1.35 through 2.0.5 2.0.5
  1.35.1 Available under license
1.36 core-js 2.6.11 2.6.11
  1.36.1 Available under license
1.37 unicode-match-property-value-ecmascript 2.0.0
  1.37.1 Available under license
1.38 lodash-repeat 4.1.0 4.1.0
  1.38.1 Available under license
1.39 babel-plugin-transform-async-to-generator 6.24.1
  1.39.1 Available under license
1.40 minimist 1.2.5 1.2.5
  1.40.1 Available under license
1.41 namespace 1.4.01
  1.41.1 Available under license
1.42 hbase-procedure 2.4.2
  1.42.1 Available under license
1.43 babel-plugin-transform-strict-mode 6.24.1 6.24.1
  1.43.1 Available under license
1.44 etag 1.7.0 1.7.0
  1.44.1 Available under license
1.45 composite-error 0.1.1 0.1.1
  1.45.1 Available under license
1.46 is-my-json-valid 2.20.5
  1.46.1 Available under license
1.47 lodash 4.17.15 4.17.15
  1.47.1 Available under license
1.48 @babel/plugin-syntax-optional-catch-binding 7.8.3
  1.48.1 Available under license
1.49 grpc-api 1.27.1
  1.49.1 Available under license
1.50 json3 3.3.2 3.3.2
  1.50.1 Available under license
1.51 jetty-xml 9.4.41.v20210516
  1.51.1 Available under license
1.52 lru-cache 4.1.1 4.1.1
  1.52.1 Available under license
1.53 groovy-groovydoc 3.0.8
  1.53.1 Available under license
1.54 qs 6.10.1 6.10.1
  1.54.1 Available under license
1.55 caniuse-lite 1.0.30001282
  1.55.1 Available under license
1.56 jackson-datatype-joda 2.4.2
  1.56.1 Available under license
1.57 options 0.0.6 0.0.6
  1.57.1 Available under license
1.58 path-exists 4.0.0
  1.58.1 Available under license
1.59 swagger-jaxrs2 2.0.7
  1.59.1 Available under license
1.60 @babel/helper-optimise-call-expression 7.16.0
  1.60.1 Available under license
1.61 abbrev 1.1.1 1.1.1
  1.61.1 Available under license
1.62 kind-of 3.2.2 3.2.2
  1.62.1 Available under license
1.63 map-age-cleaner 0.1.3 0.1.3
  1.63.1 Available under license
1.64 stax 4.2
  1.64.1 Available under license
1.65 grpc-context 1.27.1
  1.65.1 Available under license
1.66 ansi-regex 5.0.0 5.0.0
  1.66.1 Available under license
1.67 toggle-selection 1.0.6 1.0.6
  1.67.1 Available under license
1.68 es6-symbol 3.1.3
  1.68.1 Available under license
1.69 jsonpointer 4.1.0
  1.69.1 Available under license
1.70 tiny-inflate 1.0.3 1.0.3
  1.70.1 Available under license
1.71 drill-kudu-storage 1.19.0
  1.71.1 Available under license
1.72 yargs-unparser 1.6.0
  1.72.1 Available under license
1.73 source-map 0.5.7 0.5.7
1.73.1 Available under license
1.74 parquet-avro 1.8.1
1.74.1 Available under license
1.75 drill-storage-hive-core 1.19.0
1.75.1 Available under license
1.76 oas-resolver 2.5.4 2.5.4
1.76.1 Available under license
1.77 resolve-dir 1.0.1 1.0.1
1.77.1 Available under license
1.78 find-cache-dir 3.3.2
1.78.1 Available under license
1.79 helmet-csp 1.2.2 1.2.2
1.79.1 Available under license
1.80 helmet 2.3.0 2.3.0
1.80.1 Available under license
1.81 lodash.flow 3.5.0 3.5.0
1.81.1 Available under license
1.82 yargs 15.4.1
1.82.1 Available under license
1.83 form-data 2.3.3 2.3.3
1.83.1 Available under license
1.84 expiringmap 0.5.8
1.85 is-yarn-global 0.3.0
1.85.1 Available under license
1.86 checker-qual 3.5.0
1.86.1 Available under license
1.87 ansi-regex 4.1.0 4.1.0
1.87.1 Available under license
1.88 ms 2.1.2
1.88.1 Available under license
1.89 json-path 1.1.1 1.1.1
1.89.1 Available under license
1.90 hibernate-validator 5.0.0.Final
1.90.1 Available under license
1.91 jersey-media-multipart 2.5.1
1.91.1 Available under license
1.92 widest-line 3.1.0
1.92.1 Available under license
1.93 is-fullwidth-code-point 3.0.0 3.0.0
1.93.1 Available under license
1.94 netty-transport 4.1.42.Final
  1.94.1 Available under license
1.95 image-size 0.7.4 0.7.4
  1.95.1 Available under license
1.96 archiver 2.1.1 2.1.1
  1.96.1 Available under license
1.97 get-intrinsic 1.1.1
  1.97.1 Available under license
1.98 pcapngdecoder 1.2
  1.98.1 Available under license
1.99 frontline 1.3.1 1.3.1
  1.99.1 Available under license
1.100 grpc-stub 1.27.1
  1.100.1 Available under license
1.101 batik-dom 1.13
  1.101.1 Available under license
1.102 referrer-policy 1.0.0 1.0.0
  1.102.1 Available under license
1.103 commons-net 1.4.1
  1.103.1 Available under license
1.104 inquirer 0.12.0
  1.104.1 Available under license
1.105 engine.io 3.4.2 3.4.2
  1.105.1 Available under license
1.106 ipaddr.js 1.0.5 1.0.5
  1.106.1 Available under license
1.107 destroy 1.0.4 1.0.4
  1.107.1 Available under license
1.108 hive-storage-api 2.7.0
  1.108.1 Available under license
1.109 is-glob 4.0.1
  1.109.1 Available under license
1.110 istanbul-reports 3.0.2
  1.110.1 Available under license
1.111 xsdlib 2013.6.1
  1.111.1 Available under license
1.112 jetty-jmx 9.4.34.v20201102
  1.112.1 Available under license
1.113 nanoid 1.0.2 1.0.2
   1.113.1 Available under license
1.114 os-locale 3.1.0 3.1.0
   1.114.1 Available under license
1.115 @babel/helper-function-name 7.16.0
   1.115.1 Available under license
1.116 woodstox-core 6.2.0
   1.116.1 Available under license
1.117 es6-error 4.1.1
   1.117.1 Available under license
1.118 hbase-shaded-jetty 3.4.1
   1.118.1 Available under license
1.119 jsesc 0.5.0 0.5.0
   1.119.1 Available under license
1.120 libjpeg-turbo 6b
   1.120.1 Available under license
1.121 netty 4.1.59.Final
   1.121.1 Available under license
1.122 crc32-stream 2.0.0 2.0.0
   1.122.1 Available under license
1.123 binary-extensions 2.1.0 2.1.0
   1.123.1 Available under license
1.124 mongodb-driver 3.12.4
   1.124.1 Available under license
1.125 atob 2.1.2 2.1.2
   1.125.1 Available under license
1.126 zlib 1.2.11
   1.126.1 Available under license
1.127 check-error 1.0.2
   1.127.1 Available under license
1.128 jopt-simple 5.0.4
   1.128.1 Available under license
1.129 hadoop-mapreduce-client-jobclient 3.2.2
   1.129.1 Available under license
1.130 is-descriptor 1.0.2
   1.130.1 Available under license
1.131 iceberg-common 0.8.0-incubating
   1.131.1 Available under license
1.132 winston-transport 3.0.1 3.0.1
1.132.1 Available under license

1.133 **text-table 0.2.0**
   1.133.1 Available under license

1.134 **get-intrinsic 1.1.1 1.1.1**
   1.134.1 Available under license

1.135 **xmlDom 0.2.1 0.2.1**
   1.135.1 Available under license

1.136 **fast-json-stable-stringify 2.1.0 2.1.0**
   1.136.1 Available under license

1.137 **winser-with-api 1.0.1 1.0.1**
   1.137.1 Available under license

1.138 **minimatch 3.0.4 3.0.4**
   1.138.1 Available under license

1.139 **proxy-add 2.0.6 2.0.6**
   1.139.1 Available under license

1.140 **fill-range 2.2.4**
   1.140.1 Available under license

1.141 **brace 0.11.1 0.11.1**
   1.141.1 Available under license

1.142 **debug 4.3.2**
   1.142.1 Available under license

1.143 **babel-plugin-transform-es2015-for-of 6.23.0**
   1.143.1 Available under license

1.144 **okhttp 3.9.1**
   1.144.1 Available under license

1.145 **infinity-agent 1.0.2 1.0.2**
   1.145.1 Available under license

1.146 **readable-stream 2.3.6 2.3.6**
   1.146.1 Available under license

1.147 **is-buffer 1.1.6 1.1.6**
   1.147.1 Available under license

1.148 **iconv-lite 0.4.23 0.4.23**
   1.148.1 Available under license

1.149 **p-limit 2.3.0 2.3.0**
   1.149.1 Available under license

1.150 **axios 0.19.2 0.19.2**
   1.150.1 Available under license

1.151 **wtf-8 1.0.0 1.0.0**
   1.151.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>drill-format-syslog</td>
<td>1.19.0</td>
<td>Available</td>
</tr>
<tr>
<td>content-disposition</td>
<td>0.5.1</td>
<td>Available</td>
</tr>
<tr>
<td>jsreport-reports</td>
<td>2.5.1</td>
<td>Available</td>
</tr>
<tr>
<td>regenerate</td>
<td>1.4.1</td>
<td>Available</td>
</tr>
<tr>
<td>puppeteer</td>
<td>1.11.0</td>
<td>Available</td>
</tr>
<tr>
<td>drill-logical</td>
<td>1.19.0</td>
<td>Available</td>
</tr>
<tr>
<td>socket.io</td>
<td>1.7.4</td>
<td>Available</td>
</tr>
<tr>
<td>fs-extra</td>
<td>1.0.0</td>
<td>Available</td>
</tr>
<tr>
<td>commons-cli</td>
<td>1.4</td>
<td>Available</td>
</tr>
<tr>
<td>parseqs</td>
<td>0.0.5</td>
<td>Available</td>
</tr>
<tr>
<td>batik-svgpp</td>
<td>1.13</td>
<td>Available</td>
</tr>
<tr>
<td>lockfile</td>
<td>1.0.4</td>
<td>Available</td>
</tr>
<tr>
<td>groovy-nio</td>
<td>3.0.8</td>
<td>Available</td>
</tr>
<tr>
<td>leveldbjni-all</td>
<td>1.8</td>
<td>Available</td>
</tr>
<tr>
<td>batch</td>
<td>0.4.0</td>
<td>Available</td>
</tr>
<tr>
<td>bson</td>
<td>1.0.9</td>
<td>Available</td>
</tr>
<tr>
<td>isorelax</td>
<td>20090621</td>
<td>Available</td>
</tr>
<tr>
<td>objenesis</td>
<td>2.1</td>
<td>Available</td>
</tr>
<tr>
<td>bean-validation-api</td>
<td>1.1.0.Final</td>
<td>Available</td>
</tr>
<tr>
<td>xmllhttprequest-ssl</td>
<td>1.5.5</td>
<td>Available</td>
</tr>
</tbody>
</table>
1.191.1 Available under license

1.192 jakarta-annotation-api 1.3.5
   1.192.1 Available under license

1.193 netty-resolver 4.1.59.Final
   1.193.1 Available under license

1.194 babel-helper-function-name 6.24.1
   1.194.1 Available under license

1.195 apib-include-directive 0.1.0 0.1.0
   1.195.1 Available under license

1.196 raw-body 2.3.3 2.3.3
   1.196.1 Available under license

1.197 semver 5.7.1 5.7.1
   1.197.1 Available under license

1.198 json-glib 1.1.2-0ubuntu1
   1.198.1 Available under license

1.199 jetty-http 9.4.34.v20201102
   1.199.1 Available under license

1.200 @babel/parser 7.16.4
   1.200.1 Available under license

1.201 bytes 0.2.1 0.2.1
   1.201.1 Available under license

1.202 hive-hbase-handler 3.1.2
   1.202.1 Available under license

1.203 strip-ansi 3.0.1 3.0.1
   1.203.1 Available under license

1.204 invariant 2.2.4 2.2.4
   1.204.1 Available under license

1.205 async-replace 1.0.1 1.0.1
   1.205.1 Available under license

1.206 strip-ansi 0.1.1 0.1.1
   1.206.1 Available under license

1.207 jsonify 0.0.0
   1.207.1 Available under license

   1.208.1 Available under license

1.209 form-data 2.3.2 2.3.2
   1.209.1 Available under license

1.210 apache-log4j 2.17.1
   1.210.1 Available under license
1.211 clone-response 1.0.2
   1.211.1 Available under license
1.212 hbase-server 2.4.2
   1.212.1 Available under license
1.213 jython 2.7.1
   1.213.1 Available under license
1.214 htmlparser2 3.9.2 3.9.2
   1.214.1 Available under license
1.215 netty-codec 4.0.48.Final
   1.215.1 Available under license
1.216 jline 2.14.3
1.217 javassist 3.22.0-GA
   1.217.1 Available under license
1.218 google-http-client-xml 1.20.0
   1.218.1 Available under license
1.219 ini 2.0.0
   1.219.1 Available under license
1.220 @babel/plugin-proposal-aspect-generator-functions 7.16.4
   1.220.1 Available under license
1.221 jju 1.2.1 1.2.1
   1.221.1 Available under license
1.222 binary-extensions 2.1.0
   1.222.1 Available under license
1.223 junit 4.13.2
   1.223.1 Available under license
1.224 batik-svgbrowser 1.13
   1.224.1 Available under license
1.225 vm2 3.8.3 3.8.3
   1.225.1 Available under license
1.226 ip-address 5.8.9 5.8.9
   1.226.1 Available under license
1.227 cache-api 1.0.0
   1.227.1 Available under license
1.228 string-width 4.2.2 4.2.2
   1.228.1 Available under license
1.229 os-locale 1.4.0 1.4.0
   1.229.1 Available under license
1.230 cliui 6.0.0
   1.230.1 Available under license
1.231 dot-prop 5.3.0
   1.231.1 Available under license
1.232 jetty-servlet 9.3.27.v20190418
1.233 xml2js 0.4.23 0.4.23
   1.233.1 Available under license
1.234 protostuff-api 1.7.1
   1.234.1 Available under license
1.235 json-schema 0.2.3 0.2.3
   1.235.1 Available under license
1.236 require-directory 2.1.1 2.1.1
   1.236.1 Available under license
1.237 semver 7.3.5
   1.237.1 Available under license
1.238 jersey-client 2.32
   1.238.1 Available under license
1.239 jquery 3.6.0 3.6.0
   1.239.1 Available under license
1.240 slf4j-log4j 2.14.0
   1.240.1 Available under license
1.241 1to2 1.0.0 1.0.0
   1.241.1 Available under license
1.242 babel-code-frame 6.26.0 6.26.0
   1.242.1 Available under license
1.243 strip-bom 3.0.0
   1.243.1 Available under license
1.244 safer-buffer 2.1.2 2.1.2
   1.244.1 Available under license
1.245 objenesis 2.5.1
   1.245.1 Available under license
1.246 jetty-io 9.4.41.v20210516
   1.246.1 Available under license
1.247 normalize-path 3.0.0 3.0.0
   1.247.1 Available under license
1.248 flagged-respawm 1.0.1 1.0.1
   1.248.1 Available under license
1.249 slash 2.0.0
   1.249.1 Available under license
1.250 jnr-posix 3.0.27
   1.250.1 Available under license
Open Source Used in DNAC Platform Arica

1.251 validator 10.11.0 10.11.0
   1.251.1 Available under license

1.252 ansi-styles 4.3.0 4.3.0
   1.252.1 Available under license

1.253 z-schema 3.25.1 3.25.1
   1.253.1 Available under license

1.254 babel-runtime 6.26.0
   1.254.1 Available under license

1.255 has-symbols 1.0.1
   1.255.1 Available under license

1.256 is-negative-zero 2.0.0
   1.256.1 Available under license

1.257 xmlbuilder 8.2.2 8.2.2
   1.257.1 Available under license

1.258 ua-parser-js 0.7.18 0.7.18
   1.258.1 Available under license

1.259 parseuri 0.0.5 0.0.5
   1.259.1 Available under license

1.260 rxjs 6.6.3 6.6.3
   1.260.1 Available under license

1.261 babel-plugin-proposal-nullish-coalescing-operator 7.16.0
   1.261.1 Available under license

1.262 domutils 1.7.0 1.7.0
   1.262.1 Available under license

1.263 @babel/helper-validator-identifier 7.10.4
   1.263.1 Available under license

1.264 drill-format-spss 1.19.0
   1.264.1 Available under license

1.265 @babel/helper-define-polyfill-provider 0.3.0
   1.265.1 Available under license

1.266 es-abstract 1.18.0
   1.266.1 Available under license

1.267 commons-beanutils 1.9.4
   1.267.1 Available under license

1.268 socket.io 2.3.0 2.3.0
   1.268.1 Available under license

1.269 esacpe 3.6.0
   1.269.1 Available under license

1.270 ws 7.3.1 7.3.1
1.270.1 Available under license
1.271 json-stable-stringify 1.0.1
   1.271.1 Available under license
1.272 wrap-ansi 5.1.0
   1.272.1 Available under license
1.273 intl-format-cache 2.1.0 2.1.0
   1.273.1 Available under license
1.274 user-home 1.1.1
   1.274.1 Available under license
1.275 libflac 1.3.1
   1.275.1 Available under license
1.276 jetty-util-ajax 9.4.34.v20201102
   1.276.1 Available under license
1.277 eslint-module-utils 2.7.1
   1.277.1 Available under license
1.278 @babel/plugin-transform-modules-commonjs 7.16.0
   1.278.1 Available under license
1.279 aggregate-error 3.1.0
   1.279.1 Available under license
1.280 supports-color 7.2.0
   1.280.1 Available under license
1.281 fresh 0.5.2 0.5.2
   1.281.1 Available under license
1.282 intl-relativeformat 2.1.0 2.1.0
   1.282.1 Available under license
1.283 groovy-testng 3.0.8
   1.283.1 Available under license
1.284 lodash 4.17.10 4.17.10
   1.284.1 Available under license
1.285 kerb-crypto 1.0.0
   1.285.1 Available under license
1.286 parseurl 1.3.3 1.3.3
   1.286.1 Available under license
1.287 regenerator-runtime 0.13.9
   1.287.1 Available under license
1.288 jquery 3.5.1 3.5.1
   1.288.1 Available under license
1.289 bitsyntax 0.0.4 0.0.4
   1.289.1 Available under license
1.290 underscore 1.13.1 1.13.1
   1.290.1 Available under license
1.291 babel-register 6.26.0 6.26.0
   1.291.1 Available under license
1.292 strip-bom-buf 2.0.0 2.0.0
   1.292.1 Available under license
1.293 neo-async 2.6.2 2.6.2
   1.293.1 Available under license
1.294 isemail 1.2.0 1.2.0
   1.294.1 Available under license
1.295 batik-svggen 1.13
   1.295.1 Available under license
1.296 commander 1.2.0 1.2.0
   1.296.1 Available under license
1.297 lodash-union 4.6.0 4.6.0
   1.297.1 Available under license
1.298 mongodb 2.2.36 2.2.36
   1.298.1 Available under license
1.299 antlr-runtime 4.8-1
   1.299.1 Available under license
1.300 babel-plugin-polyfill-corejs3 0.4.0
   1.300.1 Available under license
1.301 keygrip 1.0.3 1.0.3
   1.301.1 Available under license
1.302 entities 1.1.1 1.1.1
   1.302.1 Available under license
1.303 rimraf 2.7.1 2.7.1
   1.303.1 Available under license
1.304 read-all-stream 1.0.2 1.0.2
   1.304.1 Available under license
1.305 harfbuzz 1.0.1-1ubuntu0.1
   1.305.1 Available under license
1.306 base64-js 1.3.1 1.3.1
   1.306.1 Available under license
1.307 home-or-tmp 2.0.0 2.0.0
   1.307.1 Available under license
1.308 de.huxhorn.lilith.sender 0.9.44
   1.308.1 Available under license
1.309 graceful-fs 4.1.11 4.1.11
1.309.1 Available under license
1.310 lodash-difference 4.5.0 4.5.0
1.310.1 Available under license
1.311 strict-uri-encode 2.0.0 2.0.0
1.311.1 Available under license
1.312 path-root 0.1.1 0.1.1
1.312.1 Available under license
1.313 es6-symbol 3.1.1
1.313.1 Available under license
1.314 typedarray-to-buffer 3.1.5
1.314.1 Available under license
1.315 jackson-annotations 2.12.1
1.315.1 Available under license
1.316 is-plain-object 2.0.4 2.0.4
1.316.1 Available under license
1.317 jackson-module-paranamer 2.10.5
1.317.1 Available under license
1.318 has-value 0.3.1 0.3.1
1.318.1 Available under license
1.319 safe-regex 1.1.0 1.1.0
1.319.1 Available under license
1.320 lodash-map 4.6.0 4.6.0
1.320.1 Available under license
1.321 color-name 1.1.4 1.1.4
1.321.1 Available under license
1.322 is-core-module 2.8.0
1.322.1 Available under license
1.323 jersey-container-servlet 2.5.1
1.323.1 Available under license
1.324 through 2.3.8 2.3.8
1.324.1 Available under license
1.325 source-map 0.6.0 0.6.0
1.325.1 Available under license
1.326 hadoop-distcp 3.2.2
1.326.1 Available under license
1.327 libsoup 2.52.2-1ubuntu0.3
1.327.1 Available under license
1.328 python-setuptools 28.8.0
1.328.1 Available under license
1.329 junit-platform-launcher 1.7.0
  1.329.1 Available under license

1.330 is-bbigint 1.0.1
  1.330.1 Available under license

1.331 pluralize 1.2.1
  1.331.1 Available under license

1.332 react-json-view 1.19.1 1.19.1
  1.332.1 Available under license

1.333 has-values 1.0.0 1.0.0
  1.333.1 Available under license

1.334 interpret 1.4.0 1.4.0
  1.334.1 Available under license

1.335 t-digest 3.2
  1.335.1 Available under license

1.336 chalk 2.4.1
  1.336.1 Available under license

1.337 summernote 0.8.15 0.8.15
  1.337.1 Available under license

1.338 avatica-metrics 1.15.0
  1.338.1 Available under license

1.339 append-transform 2.0.0
  1.339.1 Available under license

1.340 jsreport-data 2.4.0 2.4.0
  1.340.1 Available under license

1.341 escalate 3.1.1
  1.341.1 Available under license

1.342 chalk 4.1.2
  1.342.1 Available under license

1.343 css-what 2.1.3 2.1.3
  1.343.1 Available under license

1.344 react-redux 5.0.7 5.0.7
  1.344.1 Available under license

1.345 swagger-annotations 2.0.7
  1.345.1 Available under license

1.346 commons-lang3 2.6
  1.346.1 Available under license

1.347 batik-css 1.13
  1.347.1 Available under license

1.348 find-up 2.1.0
<table>
<thead>
<tr>
<th>Version</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.348.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.349</td>
<td>core 3.1.1</td>
<td></td>
</tr>
<tr>
<td>1.349.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.350</td>
<td>base64-arraybuffer 0.1.5</td>
<td></td>
</tr>
<tr>
<td>1.350.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.351</td>
<td>stream-shift 1.0.1 1.0.1</td>
<td></td>
</tr>
<tr>
<td>1.351.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.352</td>
<td>jackson-dataformat-hocon 1.1.1</td>
<td></td>
</tr>
<tr>
<td>1.352.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.353</td>
<td>jackson 1.0.0</td>
<td></td>
</tr>
<tr>
<td>1.353.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.354</td>
<td>commons-lang3 3.11</td>
<td></td>
</tr>
<tr>
<td>1.354.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.355</td>
<td>zlib 1.2.1</td>
<td></td>
</tr>
<tr>
<td>1.355.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.356</td>
<td>jackson-dataformat-yaml 2.9.9</td>
<td></td>
</tr>
<tr>
<td>1.356.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.357</td>
<td>for-in 1.0.2 1.0.2</td>
<td></td>
</tr>
<tr>
<td>1.357.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.358</td>
<td>nan 1.3.0 1.3.0</td>
<td></td>
</tr>
<tr>
<td>1.358.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.359</td>
<td>react-textarea-autosize 6.1.0 6.1.0</td>
<td></td>
</tr>
<tr>
<td>1.359.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.360</td>
<td>uri-js 4.4.0 4.4.0</td>
<td></td>
</tr>
<tr>
<td>1.360.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.361</td>
<td>call-bind 1.0.2 1.0.2</td>
<td></td>
</tr>
<tr>
<td>1.361.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.362</td>
<td>configstore 5.0.1</td>
<td></td>
</tr>
<tr>
<td>1.362.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.363</td>
<td>msgpack 0.6.6</td>
<td></td>
</tr>
<tr>
<td>1.363.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.364</td>
<td>color-name 1.1.4</td>
<td></td>
</tr>
<tr>
<td>1.364.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.365</td>
<td>jpam 1.1</td>
<td></td>
</tr>
<tr>
<td>1.365.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.366</td>
<td>hbase-shaded-gson 3.4.1</td>
<td></td>
</tr>
<tr>
<td>1.366.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.367</td>
<td>glib 2.48.2 0ubuntu4.4</td>
<td></td>
</tr>
<tr>
<td>1.367.1</td>
<td>Available under license</td>
<td></td>
</tr>
</tbody>
</table>
1.368 ical4j 1.0.2
  1.368.1 Available under license
1.369 is-extendable 0.1.1 0.1.1
  1.369.1 Available under license
1.370 h2 1.4.200
  1.370.1 Available under license
1.371 commons-compress 1.2.2 1.2.2
  1.371.1 Available under license
1.372 prompt-tmp 1.0.0 1.0.0
  1.372.1 Available under license
1.373 cypress 1.0.0 1.0.0
  1.373.1 Available under license
1.374 jackson-annotations 2.12.3
  1.374.1 Available under license
1.375 color-convert 2.0.1 2.0.1
  1.375.1 Available under license
1.376 zip-stream 2.1.3 2.1.3
  1.376.1 Available under license
1.377 istanbul-lib-instrument 4.0.3
  1.377.1 Available under license
1.378 @babel/plugin-transform-unicode-regex 7.16.0
  1.378.1 Available under license
1.379 jooq 3.13.1
  1.379.1 Available under license
1.380 nconf 0.10.0 0.10.0
  1.380.1 Available under license
1.381 swagger-core 1.5.0
1.382 deep-is 0.1.3
  1.382.1 Available under license
1.383 estraverse 5.2.0
  1.383.1 Available under license
1.384 drill-jdbc-storage 1.19.0
  1.384.1 Available under license
1.385 swagger-jersey2-jaxrs 1.5.0
1.386 extend 3.0.2 3.0.2
  1.386.1 Available under license
1.387 jsreport-pdfjs 0.5.1 0.5.1
  1.387.1 Available under license
1.388 express 4.17.1 4.17.1
1.388.1 Available under license

1.389 redux-act 1.7.4 1.7.4
1.389.1 Available under license

1.390 braces 1.8.5
1.390.1 Available under license

1.391 sparse-bittfield 3.0.3 3.0.3
1.391.1 Available under license

1.392 babel-helpers 6.24.1 6.24.1
1.392.1 Available under license

1.393 joda-time 2.9.1
1.393.1 Available under license

1.394 parquet-column 1.8.1
1.394.1 Available under license

1.395 levn 0.3.0
1.395.1 Available under license

1.396 keyv 3.1.0
1.396.1 Available under license

1.397 ms 2.1.2 2.1.2
1.397.1 Available under license

1.398 groovy-json 3.0.8
1.398.1 Available under license

1.399 aws-java-sdk-kms 1.11.704
1.399.1 Available under license

1.400 is-utf8 0.2.1 0.2.1
1.400.1 Available under license

1.401 color-convert 1.9.3 1.9.3
1.401.1 Available under license

1.402 gir1.2-gdkpixbuf-2.0 2.32.2-1ubuntu1.6
1.402.1 Available under license

1.403 regjsgen 0.5.2
1.403.1 Available under license

1.404 javax-el 3.0.0
1.404.1 Available under license

1.405 path-parse 1.0.6
1.405.1 Available under license

1.406 transaction-api 1.1
1.406.1 Available under license

1.407 jackson-dataformat-xml 2.10.4
1.407.1 Available under license
1.408 netty-buffer 4.0.48.Final
  1.408.1 Available under license
1.409 qs 4.0.0 4.0.0
  1.409.1 Available under license
1.410 send 0.13.1 0.13.1
  1.410.1 Available under license
1.411 groovy-docgenerator 3.0.8
  1.411.1 Available under license
1.412 path-loader 1.0.10 1.0.10
  1.412.1 Available under license
1.413 type-check 0.3.2 0.3.2
  1.413.1 Available under license
1.414 path-is-inside 1.0.2
  1.414.1 Available under license
1.415 has-value 1.0.0 1.0.0
  1.415.1 Available under license
1.416 p-try 1.0.0
  1.416.1 Available under license
1.417 resolve-from 2.0.0 2.0.0
  1.417.1 Available under license
1.418 guava 19.0
1.419 ws 1.1.1 1.1.1
  1.419.1 Available under license
1.420 object-inspect 1.11.0
  1.420.1 Available under license
1.421 p-locate 4.1.0
  1.421.1 Available under license
1.422 slf4j-log4j 2.8.2
  1.422.1 Available under license
1.423 mime-pull 1.9.6
  1.423.1 Available under license
1.424 hbase-shaded-netty 3.4.1
  1.424.1 Available under license
1.425 sac 1.3
  1.425.1 Available under license
1.426 hadoop-core 1.2.1
  1.426.1 Available under license
1.427 babel-plugin-transform-decorators 6.24.1
  1.427.1 Available under license
1.428 is-finite 1.1.0
   1.428.1 Available under license
1.429 debug 2.6.9 2.6.9
   1.429.1 Available under license
1.430 indexof 0.0.1 0.0.1
1.431 @babel/traverse 7.16.3
   1.431.1 Available under license
1.432 jersey-client 2.5.1
   1.432.1 Available under license
1.433 boost 1.60
   1.433.1 Available under license
1.434 boxen 5.1.2
   1.434.1 Available under license
   1.435.1 Available under license
1.436 bson 3.12.4
   1.436.1 Available under license
1.437 moment 2.27.0 2.27.0
   1.437.1 Available under license
1.438 ansi-styles 1.0.0 1.0.0
   1.438.1 Available under license
1.439 memory 0.9.0
   1.439.1 Available under license
1.440 http-errors 1.3.1 1.3.1
   1.440.1 Available under license
1.441 twill-common 0.6.0-incubating
   1.441.1 Available under license
1.442 path-exists 3.0.0
   1.442.1 Available under license
1.443 @babel/plugin-proposal-object-rest-spread 7.16.0
   1.443.1 Available under license
1.444 camelize 1.0.0 1.0.0
   1.444.1 Available under license
1.445 @babel/helper-annotate-as-pure 7.16.0
   1.445.1 Available under license
1.446 engine.io-parser 1.3.2 1.3.2
   1.446.1 Available under license
1.447 @babel/plugin-transform-literals 7.16.0
   1.447.1 Available under license
1.448 curator-client 5.1.0
   1.448.1 Available under license
1.449 babel-helper-create-regexp-features-plugin 7.16.0
   1.449.1 Available under license
1.450 hbase-shaded-miscellaneous 3.4.1
   1.450.1 Available under license
1.451 babel-plugin-transform-es2015-block-scoped-functions 6.22.0
   1.451.1 Available under license
1.452 fast-infoset 1.2.15
1.453 jkeylockmanager 2.1.0
   1.453.1 Available under license
1.454 babel-plugin-transform-async-to-generator 6.24.1 6.24.1
   1.454.1 Available under license
1.455 amqplib 0.5.2 0.5.2
   1.455.1 Available under license
1.456 get-value 2.0.6 2.0.6
   1.456.1 Available under license
1.457 lodash.reduce 4.6.0 4.6.0
   1.457.1 Available under license
1.458 open-csv 2.3
   1.458.1 Available under license
1.459 de.huxhorn.lilith.logback.classic 0.9.44
   1.459.1 Available under license
1.460 xtend 4.0.2 4.0.2
   1.460.1 Available under license
1.461 next-tick 1.0.0
   1.461.1 Available under license
1.462 glew 1.3.1-1ubuntu0.16.04.2
   1.462.1 Available under license
1.463 cliui 5.0.0 5.0.0
   1.463.1 Available under license
1.464 tephra-core 0.6.0
   1.464.1 Available under license
1.465 drill-java-exec 1.19.0
   1.465.1 Available under license
1.466 jws 3.2.2 3.2.2
   1.466.1 Available under license
1.467 finalhandler 1.1.1 1.1.1
   1.467.1 Available under license
1.468 ignore 3.3.10
   1.468.1 Available under license
1.469 is-dotfile 1.0.3
   1.469.1 Available under license
1.470 chardet 0.7.0 0.7.0
   1.470.1 Available under license
1.471 swagger-jaxrs 1.5.0
1.472 extsprintf 1.4.0 1.4.0
   1.472.1 Available under license
1.473 react-copy-to-clipboard 5.0.2 5.0.2
   1.473.1 Available under license
1.474 @babel/plugin-transform-function-name 7.16.0
   1.474.1 Available under license
1.475 spawn-wrap 2.0.0
   1.475.1 Available under license
1.476 icu 63
   1.476.1 Available under license
1.477 okio 2.8.0
   1.477.1 Available under license
1.478 buffer-indexof-polyfill 1.0.2 1.0.2
   1.478.1 Available under license
1.479 react 16.4.2 16.4.2
   1.479.1 Available under license
1.480 update-notifier 5.1.0
   1.480.1 Available under license
1.481 netty-codec 4.1.42.Final
   1.481.1 Available under license
1.482 js-tokens 4.0.0
   1.482.1 Available under license
1.483 lucene-analyzers-common 7.4.0
   1.483.1 Available under license
1.484 jaxb-runtime 2.3.1
1.485 classgraph 4.6.32
1.486 netty-handler 4.1.42.Final
   1.486.1 Available under license
1.487 arr-diff 2.0.0
   1.487.1 Available under license
1.488 has-ansi 2.0.0 2.0.0
   1.488.1 Available under license
1.489 es-abstract 1.17.7
  1.489.1 Available under license
1.490 drill-format-esri 1.19.0
  1.490.1 Available under license
1.491 lodash.isstring 4.0.1 4.0.1
  1.491.1 Available under license
1.492 amqplib 0.5.6 0.5.6
  1.492.1 Available under license
1.493 forwarded 0.1.2 0.1.2
  1.493.1 Available under license
1.494 home-or-tmp 2.0.0
  1.494.1 Available under license
1.495 mkdirp 0.5.5 0.5.5
  1.495.1 Available under license
1.496 snake-yaml 1.26
  1.496.1 Available under license
1.497 regenerate 1.4.2
  1.497.1 Available under license
1.498 json-schema-traverse 0.3.1 0.3.1
  1.498.1 Available under license
1.499 ansi-regex 3.0.0 3.0.0
  1.499.1 Available under license
1.500 jsreport-version-control 1.6.0 1.6.0
  1.500.1 Available under license
1.501 jffi 1.2.16
  1.501.1 Available under license
1.502 fs 0.0.2 0.0.2
  1.502.1 Available under license
1.503 escodegen 1.8.1
  1.503.1 Available under license
1.504 supports-color 5.5.0
  1.504.1 Available under license
1.505 jsreport 2.11.0 2.11.0
  1.505.1 Available under license
1.506 loose-envify 1.4.0 1.4.0
  1.506.1 Available under license
1.507 chrome-page-eval 1.3.0 1.3.0
  1.507.1 Available under license
1.508 ant-junit 1.10.9
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>commons-net</td>
<td>3.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-plugin-transform-async-generator-functions</td>
<td>6.24.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>cli-width</td>
<td>2.2.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>form-data</td>
<td>1.0.0-rc4</td>
<td>Available under license</td>
</tr>
<tr>
<td>content-type</td>
<td>1.0.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>libxtst</td>
<td>1.2.2-1</td>
<td>Available under license</td>
</tr>
<tr>
<td>yaml</td>
<td>1.10.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>formidable</td>
<td>1.2.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>unbox-primitive</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>calcite-avatica</td>
<td>1.17.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>micromatch</td>
<td>3.1.10</td>
<td>Available under license</td>
</tr>
<tr>
<td>tough-cookie</td>
<td>2.5.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>swagger-integration</td>
<td>2.0.7</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/plugin-proposal-logical-assignment-operators</td>
<td>7.16.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>http-signature</td>
<td>1.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>byte-buddy</td>
<td>1.9.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>proj4j</td>
<td>0.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>nss</td>
<td>3.28.4-0ubuntu0.16.04.10</td>
<td>Available under license</td>
</tr>
<tr>
<td>xml2js-preserve-spaces</td>
<td>0.0.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.528 flat-cache 1.3.4
   1.528.1 Available under license
1.529 y18n 4.0.0 4.0.0
   1.529.1 Available under license
1.530 snappy-java 1.1.2.6
1.531 estraverse 4.3.0
   1.531.1 Available under license
1.532 semver 5.5.0 5.5.0
   1.532.1 Available under license
1.533 commons-digester 2.1
   1.533.1 Available under license
1.534 @babel/helper-split-export-declaration 7.11.0
   1.534.1 Available under license
1.535 libevent 1.3.6
   1.535.1 Available under license
1.536 object-assign 4.1.1 4.1.1
   1.536.1 Available under license
1.537 compress-lzf 1.1
   1.537.1 Available under license
1.538 drill-udfs 1.19.0
   1.538.1 Available under license
1.539 is-negative-zero 2.0.1
   1.539.1 Available under license
1.540 regenerator-transform 0.10.1 0.10.1
   1.540.1 Available under license
1.541 @babel/plugin-proposal-optional-catch-binding 7.16.0
   1.541.1 Available under license
1.542 @babel/plugin-transform-modules-amd 7.16.0
   1.542.1 Available under license
1.543 path-exists 4.0.0 4.0.0
   1.543.1 Available under license
1.544 lru-cache 6.0.0
   1.544.1 Available under license
1.545 grpc-core 1.27.1
   1.545.1 Available under license
1.546 cookie-parser 1.4.3 1.4.3
   1.546.1 Available under license
1.547 swagger-annotations 1.6.2
   1.547.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-key</td>
<td>3.1.1</td>
</tr>
<tr>
<td>rimraf</td>
<td>3.0.2</td>
</tr>
<tr>
<td>babel-core</td>
<td>6.26.3</td>
</tr>
<tr>
<td>mime-types</td>
<td>2.1.30</td>
</tr>
<tr>
<td>progress</td>
<td>2.0.3</td>
</tr>
<tr>
<td>okhttp</td>
<td>2.7.5</td>
</tr>
<tr>
<td>is-ci</td>
<td>2.0.0</td>
</tr>
<tr>
<td>glob-parent</td>
<td>2.0.0</td>
</tr>
<tr>
<td>ansi-styles</td>
<td>2.2.1</td>
</tr>
<tr>
<td>axios</td>
<td>0.18.1</td>
</tr>
<tr>
<td>activation-api</td>
<td>1.1</td>
</tr>
<tr>
<td>jnr-ffi</td>
<td>2.1.7</td>
</tr>
<tr>
<td>hadoop-aws</td>
<td>3.2.2</td>
</tr>
<tr>
<td>parse-glob</td>
<td>3.0.4</td>
</tr>
<tr>
<td>supports-color</td>
<td>5.5.0</td>
</tr>
<tr>
<td>form-data</td>
<td>2.5.0</td>
</tr>
<tr>
<td>yauaa</td>
<td>5.20</td>
</tr>
<tr>
<td>batik-awt-util</td>
<td>1.13</td>
</tr>
<tr>
<td>pi-spec-converter-web</td>
<td>0.0.0</td>
</tr>
<tr>
<td>@babel/runtime</td>
<td>7.16.3</td>
</tr>
</tbody>
</table>
1.567.1 Available under license

1.568 async-each 1.0.3
1.568.1 Available under license

1.569 moment-timezone 0.5.33 0.5.33
1.569.1 Available under license

1.570 amdefine 1.0.1
1.570.1 Available under license

1.571 unicode-match-property-ecmascript 2.0.0
1.571.1 Available under license

1.572 import-lazy 2.1.0
1.572.1 Available under license

1.573 is-buffer 1.1.6
1.573.1 Available under license

1.574 package-hash 4.0.0
1.574.1 Available under license

1.575 v8flags 3.2.0
1.575.1 Available under license

1.576 string-width 4.2.3
1.576.1 Available under license

1.577 socket.io-adapter 1.1.2 1.1.2
1.577.1 Available under license

1.578 ajv-keywords 1.5.1
1.578.1 Available under license

1.579 get-func-name 2.0.0
1.579.1 Available under license

1.580 jsreport-express 2.8.1 2.8.1
1.580.1 Available under license

1.581 httpasyncclient 4.1.4
1.581.1 Available under license

1.582 hogan 3.0.2 3.0.2
1.582.1 Available under license

1.583 hikaricp 3.4.2
1.583.1 Available under license

1.584 chokidar 3.5.2
1.584.1 Available under license

1.585 logback-core 1.2.3
1.585.1 Available under license

1.586 mime 1.3.4 1.3.4
1.586.1 Available under license
1.587 jline-builtin 3.12.1
1.587.1 Available under license

1.588 orc-core 1.5.1
1.588.1 Available under license

1.589 failureaccess 1.0.1
1.589.1 Available under license

1.590 commons-crypto 1.0.0
1.591 htrace-core4 4.1.0-incubating
1.591.1 Available under license

1.592 debug 4.3.1
1.592.1 Available under license

1.593 is 3.3.0 3.3.0
1.593.1 Available under license

1.594 istanbul-lib-report 3.0.0
1.594.1 Available under license

1.595 json-schema-ref-parser 3.3.1 3.3.1
1.595.1 Available under license

1.596 babel-plugin-transform-es2015-shorthand-properties 6.24.1 6.24.1
1.596.1 Available under license

1.597 re2j 1.1
1.598 aws4 1.10.1 1.10.1
1.598.1 Available under license

1.599 clone-deep 4.0.1
1.599.1 Available under license

1.600 chai-http 3.0.0
1.600.1 Available under license

1.601 lcms 2.6 3ubuntu2.1
1.601.1 Available under license

1.602 stack-trace 0.0.10 0.0.10
1.602.1 Available under license

1.603 jcommander 1.30
1.603.1 Available under license

1.604 content-disposition 0.5.2 0.5.2
1.604.1 Available under license

1.605 stealthy-require 1.1.1 1.1.1
1.605.1 Available under license

1.606 node-preload 0.2.1
1.606.1 Available under license

1.607 sqlite 3.21.0
1.627 cliui 3.2.0 3.2.0
   1.627.1 Available under license
1.628 ms 0.7.1 0.7.1
   1.628.1 Available under license
1.629 lucene-sandbox 7.4.0
   1.629.1 Available under license
1.630 json5 2.2.0
   1.630.1 Available under license
1.631 es5-ext 0.10.53
   1.631.1 Available under license
1.632 commons-digester 1.8
1.633 commons-cli 1.2
   1.633.1 Available under license
1.634 pinkie 2.0.4 2.0.4
   1.634.1 Available under license
1.635 lodash.isnil 4.0.0 4.0.0
   1.635.1 Available under license
1.636 callsites 0.2.0
   1.636.1 Available under license
1.637 intl-messageformat-parser 1.4.0 1.4.0
   1.637.1 Available under license
1.638 std 4.1
   1.638.1 Available under license
1.639 ee-first 1.1.1 1.1.1
   1.639.1 Available under license
1.640 path-to-regexp 1.8.0 1.8.0
   1.640.1 Available under license
1.641 mime-lookup 0.0.2 0.0.2
   1.641.1 Available under license
1.642 globals 9.18.0
   1.642.1 Available under license
1.643 iconv-lite 0.4.19 0.4.19
   1.643.1 Available under license
1.644 dashify 0.2.2 0.2.2
   1.644.1 Available under license
1.645 babel-helper-hoist-variables 6.24.1
   1.645.1 Available under license
1.646 ipaddr.js 1.8.0 1.8.0
   1.646.1 Available under license
1.647 fs-constants 1.0.0 1.0.0
  1.647.1 Available under license
1.648 isbinaryfile 3.0.3 3.0.3
  1.648.1 Available under license
1.649 @babel/plugin-transform-named-capturing-groups-regex 7.16.0
  1.649.1 Available under license
1.650 babylon 6.18.0
  1.650.1 Available under license
1.651 swagger-annotations 1.5.16
  1.651.1 Available under license
1.652 lodash-bind 4.2.1 4.2.1
  1.652.1 Available under license
1.653 hk2-utils 2.2.0-b21
  1.653.1 Available under license
1.654 swagger 2.0.7
  1.654.1 Available under license
1.655 protobuf-java 3.14.0
  1.655.1 Available under license
1.656 pend 1.2.0 1.2.0
  1.656.1 Available under license
1.657 llvm 6.0-1ubuntu2~16.04.1
  1.657.1 Available under license
1.658 commons-lang3 3.8.1
  1.658.1 Available under license
1.659 math-random 1.0.4
  1.659.1 Available under license
1.660 animal-sniffer-annotation 1.18
1.661 homedir-polyfill 1.0.3 1.0.3
  1.661.1 Available under license
  1.662.1 Available under license
1.663 depd 1.1.0 1.1.0
  1.663.1 Available under license
1.664 string_decoder 1.0.3 1.0.3
  1.664.1 Available under license
1.665 isobject 3.0.1 3.0.1
  1.665.1 Available under license
1.666 object-keys 1.0.12
  1.666.1 Available under license
1.667 resolve-from 1.0.1
   1.667.1 Available under license
1.668 string.prototype.trimStart 1.0.4
   1.668.1 Available under license
1.669 braces 2.3.2 2.3.2
   1.669.1 Available under license
1.670 object-defaults 1.1.0 1.1.0
   1.670.1 Available under license
1.671 commons-math 2.2
   1.671.1 Available under license
1.672 nspr 4.13.1 0ubuntu0.16.04.1
   1.672.1 Available under license
1.673 jsreport-authorization 2.4.0 2.4.0
   1.673.1 Available under license
1.674 esutils 2.0.3 2.0.3
   1.674.1 Available under license
1.675 yauaa-logparser 5.20
   1.675.1 Available under license
1.676 @babel/helper-member-expression-to-functions 7.16.0
   1.676.1 Available under license
1.677 prelude-ls 1.1.2 1.1.2
   1.677.1 Available under license
1.678 protobuf-java 3.11.1
   1.678.1 Available under license
1.679 strip-eof 1.0.0 1.0.0
   1.679.1 Available under license
1.680 anymatch 1.3.2
   1.680.1 Available under license
1.681 @babel/runtime 7.12.1 7.12.1
   1.681.1 Available under license
1.682 lodash.once 4.1.1 4.1.1
   1.682.1 Available under license
1.683 shebang-command 2.0.0
   1.683.1 Available under license
1.684 glob-parent 5.1.2
   1.684.1 Available under license
1.685 extglob 2.0.4 2.0.4
   1.685.1 Available under license
1.686 jsp 2.3.1
1.686.1 Available under license
1.687 dns-prefetch-control 0.1.0 0.1.0
  1.687.1 Available under license
1.688 pseudomap 1.0.2 1.0.2
  1.688.1 Available under license
1.689 babel-plugin-transform-exponentiation-operator 6.24.1
  1.689.1 Available under license
1.690 jackson-mapper-asl 1.9.13
  1.690.1 Available under license
1.691 babel-helper-regex 6.26.0 6.26.0
  1.691.1 Available under license
1.692 ansi-styles 2.2.1 2.2.1
  1.692.1 Available under license
1.693 hive-service-rpc 3.1.2
  1.693.1 Available under license
1.694 make-iterator 1.0.1 1.0.1
  1.694.1 Available under license
1.695 reactive-streams 1.0.2
  1.695.1 Available under license
1.696 bluebird 3.5.4 3.5.4
  1.696.1 Available under license
1.697 fast-csv 3.7.0 3.7.0
  1.697.1 Available under license
1.698 sketches-core 0.9.0
  1.698.1 Available under license
1.699 babel-plugin-transform-es2015-modules-systemjs 6.24.1
  1.699.1 Available under license
1.700 httpsnippet 1.16.5 1.16.5
  1.700.1 Available under license
1.701 ansi-styles 3.2.1
  1.701.1 Available under license
1.702 opentype-js 0.8.0 0.8.0
  1.702.1 Available under license
1.703 jersey-server 2.32
  1.703.1 Available under license
1.704 unescape 1.0.1 1.0.1
  1.704.1 Available under license
1.705 @babel/plugin-syntax-class-static-block 7.14.5
  1.705.1 Available under license
1.706 java-jwt 3.8.3
   1.706.1 Available under license
1.707 pure-color 1.3.0 1.3.0
   1.707.1 Available under license
1.708 js-tokens 3.0.2 3.0.2
   1.708.1 Available under license
1.709 jersey-bean-validation 2.5.1
1.710 zip-stream 1.2.0 1.2.0
   1.710.1 Available under license
1.711 cacheable-request 6.1.0
   1.711.1 Available under license
1.712 geo-ip 2.14.0
   1.712.1 Available under license
1.713 kerb-common 1.0.0
   1.713.1 Available under license
1.714 html-to-xlsx 2.0.3 2.0.3
   1.714.1 Available under license
1.715 yargs 16.2.0 16.2.0
   1.715.1 Available under license
1.716 minimist 0.0.8 0.0.8
   1.716.1 Available under license
1.717 lodash-groupby 4.6.0 4.6.0
   1.717.1 Available under license
1.718 big-integer 1.6.48 1.6.48
   1.718.1 Available under license
1.719 regexpu-core 2.0.0 2.0.0
   1.719.1 Available under license
1.720 mv 2.1.1 2.1.1
   1.720.1 Available under license
1.721 winston 2.4.5 2.4.5
   1.721.1 Available under license
1.722 netty-transport 4.1.59.Final
   1.722.1 Available under license
1.723 jsreport-studio-theme-dark 0.3.0 0.3.0
   1.723.1 Available under license
1.724 jsesc 2.5.2
   1.724.1 Available under license
1.725 drill-format-pcapng 1.19.0
   1.725.1 Available under license
1.726 node-promise 0.5.14 0.5.14
   1.726.1 Available under license
1.727 byte-buddy-agent 1.9.3
   1.727.1 Available under license
1.728 uritemplate 0.3.4 0.3.4
   1.728.1 Available under license
1.729 @babel/plugin-transform-async-to-generator 7.16.0
   1.729.1 Available under license
1.730 socket.io-client 2.3.0 2.3.0
   1.730.1 Available under license
1.731 indent-string 4.0.0
   1.731.1 Available under license
1.732 packaging 16.8
   1.732.1 Available under license
1.733 netty 4.1.53.Final
   1.733.1 Available under license
1.734 depd 1.1.1 1.1.1
   1.734.1 Available under license
1.735 jsreport-tags 2.5.0 2.5.0
   1.735.1 Available under license
1.736 normalize-path 2.1.1 2.1.1
   1.736.1 Available under license
1.737 pirates 4.0.1
   1.737.1 Available under license
1.738 os-homedir 1.0.2 1.0.2
   1.738.1 Available under license
1.739 side-channel 1.0.4 1.0.4
   1.739.1 Available under license
1.740 xtend 4.0.1 4.0.1
   1.740.1 Available under license
1.741 immer 8.0.1 8.0.1
   1.741.1 Available under license
1.742 should-type 1.4.0 1.4.0
   1.742.1 Available under license
1.743 is-typedarray 1.0.0 1.0.0
   1.743.1 Available under license
1.744 has-tostringtag 1.0.0
   1.744.1 Available under license
1.745 mem 4.3.0 4.3.0
1.745.1 Available under license
1.746 **regenerator-runtime 0.13.3 0.13.3**
    1.746.1 Available under license
1.747 **hadoop-yarn-api 3.2.2**
    1.747.1 Available under license
1.748 **ret 0.1.15 0.1.15**
    1.748.1 Available under license
1.749 **synchronous-promise 2.0.15 2.0.15**
    1.749.1 Available under license
1.750 **lodash-find 4.6.0 4.6.0**
    1.750.1 Available under license
1.751 **traverse 0.6.6 0.6.6**
    1.751.1 Available under license
1.752 **netty-buffer 4.1.59.Final**
    1.752.1 Available under license
1.753 **lodash-pickby 4.6.0**
    1.753.1 Available under license
1.754 **jackson-xc 2.10.4**
    1.754.1 Available under license
1.755 **opentype-js 1.1.0 1.1.0**
    1.755.1 Available under license
1.756 **promise 7.3.1 7.3.1**
    1.756.1 Available under license
1.757 **mongodb-driver-core 3.12.4**
    1.757.1 Available under license
1.758 **babel-plugin-transform-es2015-literals 6.22.0**
    1.758.1 Available under license
1.759 **ip-address 5.3.1**
    1.759.1 Available under license
1.760 **hadoop 1.2.1**
    1.760.1 Available under license
1.761 **glib-networking 2.48.2 1~ubuntu16.04.1**
    1.761.1 Available under license
1.762 **commons-codec 1.14**
    1.762.1 Available under license
1.763 **source-map 0.6.1 0.6.1**
    1.763.1 Available under license
1.764 **find-up 4.1.0 4.1.0**
    1.764.1 Available under license
1.765 locate-path 5.0.0
   1.765.1 Available under license
1.766 bunyan-format 0.2.1 0.2.1
   1.766.1 Available under license
1.767 es6-promise 3.3.1 3.3.1
   1.767.1 Available under license
1.768 is-string 1.0.7
   1.768.1 Available under license
1.769 json 1.8
   1.769.1 Available under license
1.770 unset-value 1.0.0 1.0.0
   1.770.1 Available under license
1.771 calcite-core 1.26.0
   1.771.1 Available under license
1.772 locid 1.0.0 1.0.0
   1.772.1 Available under license
1.773 debug 4.3.4
   1.773.1 Available under license
1.774 jcip-annotation 1.0
   1.774.1 Available under license
1.775 react-highlight-words 0.16.0 0.16.0
   1.775.1 Available under license
1.776 @babel/helper-plugin-utils 7.13.0
   1.776.1 Available under license
1.777 prop-types 15.6.2 15.6.2
   1.777.1 Available under license
1.778 parquet-format-structures 1.12.0
   1.778.1 Available under license
1.779 babel-plugin-syntax-top-level-await 7.14.5
   1.779.1 Available under license
1.780 annotations 2.0.1
   1.780.1 Available under license
1.781 sanitize-html-react 1.13.0 1.13.0
   1.781.1 Available under license
1.782 statuses 1.2.1 1.2.1
   1.782.1 Available under license
1.783 babel-plugin-transform-es2015-classes 6.24.1 6.24.1
   1.783.1 Available under license
1.784 async 1.0.0 1.0.0
<table>
<thead>
<tr>
<th>Version</th>
<th>Package Name</th>
<th>Version Information</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.784.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.785</td>
<td>agent-base 4.3.0 4.3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.785.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.786</td>
<td>camelcase 2.1.1 2.1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.786.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.787</td>
<td>pixman 0.33.6 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.787.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.788</td>
<td>drill-iceberg-metastore 1.19.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.788.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.789</td>
<td>isexe 2.0.0 2.0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.789.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.790</td>
<td>is-number 7.0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.790.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.791</td>
<td>unicode-trie 0.3.1 0.3.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.791.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.792</td>
<td>commander 4.1.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.792.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.793</td>
<td>nomnom 1.8.1 1.8.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.793.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.794</td>
<td>asn1 0.2.4 0.2.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.794.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.795</td>
<td>component-bind 1.0.0 1.0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.795.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.796</td>
<td>@babel/compat-data 7.16.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.796.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.797</td>
<td>httpcore-nio 4.4.10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.797.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.798</td>
<td>@babel/runtime 7.0.0-beta.44 7.0.0-beta.44</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.798.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.799</td>
<td>invariant 2.2.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.799.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.800</td>
<td>hk2-locator 2.2.0-b21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.800.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.801</td>
<td>json-stringify-safe 5.0.1 5.0.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.801.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.802</td>
<td>babel-plugin-transform-es2015-modules-amd 6.24.1</td>
<td>Available under license</td>
<td></td>
</tr>
<tr>
<td>1.802.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.803</td>
<td>jansi 1.18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.803.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1.804 is-data-descriptor 1.0.0
   1.804.1 Available under license
1.805 groovy-datetime 3.0.8
   1.805.1 Available under license
1.806 kerb-admin 1.0.0
   1.806.1 Available under license
1.807 http2-client 1.3.3 1.3.3
   1.807.1 Available under license
1.808 jbig-kit 2.1-3.1
   1.808.1 Available under license
1.809 hbase-replication 2.4.2
   1.809.1 Available under license
1.810 passport-http-bearer 1.0.1 1.0.1
   1.810.1 Available under license
1.811 hk2-locator 2.5.0-b32
   1.811.1 Available under license
1.812 prefixmap 2.0
   1.812.1 Available under license
1.813 buffer-from 1.1.1 1.1.1
   1.813.1 Available under license
1.814 jnr-constants 0.9.9
1.815 annotations 13.0
   1.815.1 Available under license
1.816 parquet-jackson 1.12.0
   1.816.1 Available under license
1.817 json5 0.5.1
   1.817.1 Available under license
1.818 avatica-metrics 1.17.0
   1.818.1 Available under license
1.819 lodash.escaperegexp 4.1.2 4.1.2
   1.819.1 Available under license
1.820 formidable 1.2.2
   1.820.1 Available under license
1.821 graphlib 2.1.8 2.1.8
   1.821.1 Available under license
1.822 type-is 1.6.18 1.6.18
   1.822.1 Available under license
1.823 jacoco-core 0.7.9
   1.823.1 Available under license
1.824 is-data-descriptor 1.0.0 1.0.0
   1.824.1 Available under license
1.825 delayed-stream 1.0.0 1.0.0
   1.825.1 Available under license
1.826 libxdmcp 1.1.2-1.1
   1.826.1 Available under license
1.827 caniuse-lite 1.0.30000874 1.0.30000874
   1.827.1 Available under license
1.828 pkg-dir 3.0.0
   1.828.1 Available under license
1.829 append-field 1.0.0 1.0.0
   1.829.1 Available under license
1.830 babel-plugin-dynamic-import-node 2.3.3
   1.830.1 Available under license
1.831 error_prone_annotations 2.3.4
   1.831.1 Available under license
1.832 is-extglob 1.0.0
   1.832.1 Available under license
1.833 scheduler 0.19.1 0.19.1
   1.833.1 Available under license
1.834 at-spi2-core 2.18.3-4ubuntu1
   1.834.1 Available under license
1.835 aws-java-sdk-sts 1.11.375
   1.835.1 Available under license
1.836 jersey-container-servlet-core 2.32
   1.836.1 Available under license
1.837 jsreport-public-templates 2.2.0 2.2.0
   1.837.1 Available under license
1.838 metrics-servlets 4.1.0
1.839 debug 3.2.7 3.2.7
   1.839.1 Available under license
1.840 exit-hook 1.1.1
   1.840.1 Available under license
1.841 batik-bridge 1.13
   1.841.1 Available under license
1.842 jackson 1.9.13
   1.842.1 Available under license
1.843 string_decoder 0.10.31 0.10.31
   1.843.1 Available under license
1.844 slf4j 1.7.28
  1.844.1 Available under license
1.845 readline2 1.0.1
  1.845.1 Available under license
1.846 twill-discovery-core 0.6.0-incubating
  1.846.1 Available under license
1.847 bootstrap 3.4.1 3.4.1
  1.847.1 Available under license
1.848 object-inspect 1.8.0
  1.848.1 Available under license
1.849 aws-java-sdk-core 1.11.375
  1.849.1 Available under license
1.850 calcite-linq4j 1.26.0
  1.850.1 Available under license
1.851 babel-plugin-transform-es2015-unicode-regex 6.24.1
  1.851.1 Available under license
1.852 json-path 2.4.0
  1.852.1 Available under license
1.853 netty-buffer 4.1.42.Final
  1.853.1 Available under license
1.854 acorn 3.3.0
  1.854.1 Available under license
1.855 hppc 0.7.1
  1.855.1 Available under license
1.856 log4j-api 2.17.1
  1.856.1 Available under license
1.857 js-tokens 4.0.0 4.0.0
  1.857.1 Available under license
1.858 string_decoder 1.1.1 1.1.1
  1.858.1 Available under license
1.859 node-forge-flash 0.0.0
  1.859.1 Available under license
1.860 httpcomponents-core 4.4.10
  1.860.1 Available under license
1.861 curvesapi 1.06
  1.861.1 Available under license
1.862 growl 1.10.5
  1.862.1 Available under license
1.863 classmate 0.8.0
1.863.1 Available under license

1.864 is-regex 1.1.1
1.864.1 Available under license

1.865 end-of-stream 1.4.4 1.4.4
1.865.1 Available under license

1.866 collection-visit 1.0.0 1.0.0
1.866.1 Available under license

1.867 glob-base 0.3.0
1.867.1 Available under license

1.868 react-dom 16.4.1 16.4.1
1.868.1 Available under license

1.869 babel-helper-explode-assignable-expression 6.24.1
1.869.1 Available under license

1.870 netty-codec-socks 4.1.42.Final
1.870.1 Available under license

1.871 string-prototype-trimend 1.0.1
1.871.1 Available under license

1.872 string-width 3.1.0
1.872.1 Available under license

1.873 slf4j-ext 1.7.25

1.874 jersey 2.5.1
1.874.1 Available under license

1.875 proto-google-common-protos 1.17.0
1.875.1 Available under license

1.876 batik-script 1.13
1.876.1 Available under license

1.877 curator-recipes 5.1.0
1.877.1 Available under license

1.878 kerb-core 1.0.0
1.878.1 Available under license

1.879 groovy-sql 3.0.8
1.879.1 Available under license

1.880 request 2.88.2 2.88.2
1.880.1 Available under license

1.881 ignore-by-default 1.0.1
1.881.1 Available under license

1.882 user-home 2.0.0
1.882.1 Available under license

1.883 chainsaw 0.1.0 0.1.0
1.883.1 Available under license
1.884 jsreport-licensing 2.2.3
  1.884.1 Available under license
1.885 url-parse-lax 3.0.0
  1.885.1 Available under license
1.886 cross-spawn 7.0.3
  1.886.1 Available under license
1.887 archiver 3.0.0
  1.887.1 Available under license
1.888 coffee-script 2.5.1
  1.888.1 Available under license
1.889 string-replace-async 2.0.0
  1.889.1 Available under license
1.890 browser-stdout 1.3.1
  1.890.1 Available under license
1.891 cliui 6.0.0
  1.891.1 Available under license
1.892 listener-collection 1.2.0
  1.892.1 Available under license
1.893 http-cache-semantics 4.1.0
  1.893.1 Available under license
1.894 metrics-json 4.0.2
1.895 semver 5.6.0
  1.895.1 Available under license
1.896 domelementtype 1.1.3
  1.896.1 Available under license
1.897 class-utils 0.3.6
  1.897.1 Available under license
1.898 node-environment-flags 1.0.6
  1.898.1 Available under license
1.899 icu 44
  1.899.1 Available under license
1.900 guava 22.0-android
  1.900.1 Available under license
1.901 superagent 2.3.0
  1.901.1 Available under license
1.902 should-type-adaptors 1.1.0
  1.902.1 Available under license
1.903 wrappy 1.0.2
<table>
<thead>
<tr>
<th>Version</th>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.903</td>
<td>arica</td>
<td>1.903.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.904</td>
<td>capnproto</td>
<td>0.5.3-2ubuntu1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.905</td>
<td>apiguardian-api</td>
<td>1.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.906</td>
<td>jansi-native</td>
<td>1.8</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.907</td>
<td>bufferutil</td>
<td>1.2.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.908</td>
<td>to-fast-properties</td>
<td>1.0.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.909</td>
<td>jackson-annotations</td>
<td>2.11.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.910</td>
<td>execa</td>
<td>1.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.911</td>
<td>crc32-stream</td>
<td>3.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.912</td>
<td>jsesc</td>
<td>1.3.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.913</td>
<td>guice</td>
<td>4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.914</td>
<td>common-utils</td>
<td>6.1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.915</td>
<td>jsreport-studio</td>
<td>2.10.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.916</td>
<td>binary-extensions</td>
<td>1.13.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.917</td>
<td>@babel/code-frame</td>
<td>7.5.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.918</td>
<td>ajv</td>
<td>5.5.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.919</td>
<td>whatwg-fetch</td>
<td>3.4.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.920</td>
<td>regjsparser</td>
<td>0.1.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.921</td>
<td>asm</td>
<td>2.2.0-b21</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.922</td>
<td>timed-out</td>
<td>2.0.0</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.923 babel-helper-optimise-call-expression 6.24.1
   1.923.1 Available under license
1.924 is-number 7.0.0 7.0.0
   1.924.1 Available under license
1.925 base64-arraybuffer 0.1.4 0.1.4
   1.925.1 Available under license
1.926 jackson 2.10.5
   1.926.1 Available under license
1.927 babel-plugin-transform-es2015-typeof-symbol 6.23.0 6.23.0
   1.927.1 Available under license
1.928 is-posix-bracket 0.1.1
   1.928.1 Available under license
1.929 raw-body 2.4.0 2.4.0
   1.929.1 Available under license
1.930 fstream 1.0.12 1.0.12
   1.930.1 Available under license
1.931 jython 2.5.2
   1.931.1 Available under license
1.932 okhttp 4.4.0
   1.932.1 Available under license
1.933 source-map-url 0.4.0 0.4.0
   1.933.1 Available under license
1.934 fresh 0.3.0 0.3.0
   1.934.1 Available under license
1.935 json-schema-validator 0.1.16
   1.935.1 Available under license
1.936 moment-timezone 0.5.21 0.5.21
   1.936.1 Available under license
1.937 async-replace-with-limit 1.1.0 1.1.0
   1.937.1 Available under license
1.938 escape-html 1.0.3 1.0.3
   1.938.1 Available under license
1.939 orc-shims 1.5.6
   1.939.1 Available under license
1.940 inherits 2.0.3 2.0.3
   1.940.1 Available under license
1.941 babel-messages 6.23.0
   1.941.1 Available under license
1.942 httpcore5-h 5.0.2
| 1.942 | Available under license |
| 1.943 | lodash-isfunction 3.0.9 3.0.9 |
| 1.943.1 | Available under license |
| 1.944 | is-fullwidth-code-point 3.0.0 |
| 1.944.1 | Available under license |
| 1.945 | servlet-api 3.1.0 |
| 1.945.1 | Available under license |
| 1.946 | locate-path 3.0.0 3.0.0 |
| 1.946.1 | Available under license |
| 1.947 | node-forge 0.10.0 0.10.0 |
| 1.947.1 | Available under license |
| 1.948 | rc 1.2.8 |
| 1.948.1 | Available under license |
| 1.949 | has-flag 3.0.0 3.0.0 |
| 1.949.1 | Available under license |
| 1.950 | find-up 4.1.0 |
| 1.950.1 | Available under license |
| 1.951 | socket.io-parser 3.4.1 3.4.1 |
| 1.951.1 | Available under license |
| 1.952 | de.huxhorn.sulky.io 0.9.17 |
| 1.952.1 | Available under license |
| 1.953 | ultron 1.0.2 1.0.2 |
| 1.953.1 | Available under license |
| 1.954 | call-me-maybe 1.0.1 1.0.1 |
| 1.954.1 | Available under license |
| 1.955 | cycle 1.0.3 1.0.3 |
| 1.955.1 | Available under license |
| 1.956 | jsreport-handlebars 2.1.0 2.1.0 |
| 1.956.1 | Available under license |
| 1.957 | zip 1.2.0 1.2.0 |
| 1.957.1 | Available under license |
| 1.958 | jsreport-cli 2.2.5 2.2.5 |
| 1.958.1 | Available under license |
| 1.959 | commons-lang3 3.10 |
| 1.959.1 | Available under license |
| 1.960 | emojis-list 3.0.0 3.0.0 |
| 1.960.1 | Available under license |
| 1.961 | phantom-page-eval 1.2.0 1.2.0 |
| 1.961.1 | Available under license |
1.962 backo2 1.0.2 1.0.2
   1.962.1 Available under license

1.963 moment 2.22.1 2.22.1
   1.963.1 Available under license

1.964 commons-collections 4.4
   1.964.1 Available under license

1.965 jersey-container-servlet 2.25.1
   1.965.1 Available under license

1.966 commons-codec 1.12
   1.966.1 Available under license

1.967 jmespath-java 1.11.375
   1.967.1 Available under license

1.968 nanoid 1.0.7 1.0.7
   1.968.1 Available under license

1.969 batik-shared-resources 1.13
   1.969.1 Available under license

1.970 querystringify 2.2.0 2.2.0
   1.970.1 Available under license

1.971 swagger-models 2.0.7
   1.971.1 Available under license

1.972 deep-equal 0.2.2 0.2.2
   1.972.1 Available under license

1.973 spss-reader 1.3
   1.973.1 Available under license

1.974 kind-of 5.1.0 5.1.0
   1.974.1 Available under license

1.975 guava 28.2-jre
   1.975.1 Available under license

1.976 balanced-match 1.0.0 1.0.0
   1.976.1 Available under license

1.977 lodash.isboolean 3.0.3 3.0.3
   1.977.1 Available under license

1.978 filter-obj 1.1.0 1.1.0
   1.978.1 Available under license

1.979 emoji-regex 8.0.0
   1.979.1 Available under license

1.980 babel-helper-compilation-targets 7.16.3
   1.980.1 Available under license

1.981 istanbul-lib-hook 3.0.0
1.981.1 Available under license
1.982 merge2 1.3.0 1.3.0
1.982.1 Available under license
1.983 metadata-extractor 2.13.0
1.983.1 Available under license
1.984 batik-extension 1.13
1.984.1 Available under license
1.985 jetty-util 9.4.34.v20201102
1.985.1 Available under license
1.986 boost 1.58.0+dkg 5ubuntu3.1
1.986.1 Available under license
1.987 depd 1.1.2 1.1.2
1.987.1 Available under license
1.988 netty 4.0.48.Final
1.988.1 Available under license
1.989 hide-powered-by 1.0.0 1.0.0
1.989.1 Available under license
1.990 fast-levenshtein 2.0.6
1.990.1 Available under license
1.991 fbjs 0.8.17 0.8.17
1.991.1 Available under license
1.992 is-date-object 1.0.2
1.992.1 Available under license
1.993 hk2-utils 2.6.1
1.993.1 Available under license
1.994 memoize-one 4.1.0 4.1.0
1.994.1 Available under license
1.995 object-map 1.0.1 1.0.1
1.995.1 Available under license
1.996 hadoop-annotations 3.2.2
1.996.1 Available under license
1.997 codemirror 5.58.2 5.58.2
1.997.1 Available under license
1.998 blob 0.0.5 0.0.5
1.998.1 Available under license
1.999 acorn-jsx 3.0.1
1.999.1 Available under license
1.1000 send 0.13.2 0.13.2
1.1000.1 Available under license
1.1001 uuid 3.3.3 3.3.3
  1.1001.1 Available under license

1.1002 babel-helper-get-function-arity 6.24.1
  1.1002.1 Available under license

1.1003 array-unique 0.2.1
  1.1003.1 Available under license

1.1004 stax 3.1.4
  1.1004.1 Available under license

1.1005 is-my-ip-valid 1.0.0 1.0.0
  1.1005.1 Available under license

1.1006 json-schema-faker 0.5.0-rcv.34 0.5.0-rcv.34
  1.1006.1 Available under license

1.1007 aws-java-sdk-sts 1.11.837
  1.1007.1 Available under license

1.1008 oas-linter 3.2.1 3.2.1
  1.1008.1 Available under license

1.1009 y18n 4.0.3 4.0.3
  1.1009.1 Available under license

1.1010 window-size 0.1.4 0.1.4
  1.1010.1 Available under license

  1.1011.1 Available under license

1.1012 babel-plugin-transform-es2015-object-super 6.24.1
  1.1012.1 Available under license

1.1013 @babel/helper-module-transforms 7.16.0
  1.1013.1 Available under license

1.1014 akka 2.5.31
  1.1014.1 Available under license

1.1015 netty-transport-native-epoll 4.1.59.Final
  1.1015.1 Available under license

1.1016 drill-storage-http 1.19.0
  1.1016.1 Available under license

1.1017 avro-mapred 1.7.7
  1.1017.1 Available under license

1.1018 hbase 2.4.2
  1.1018.1 Available under license

1.1019 picomatch 2.2.2
  1.1019.1 Available under license

1.1020 hpkp 1.2.0 1.2.0
<table>
<thead>
<tr>
<th>1.1020</th>
<th>Available under license</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1021</td>
<td>invert-kv 2.0.0 2.0.0</td>
</tr>
<tr>
<td></td>
<td>1.1021.1 Available under license</td>
</tr>
<tr>
<td>1.1022</td>
<td>javax-servlet-jsp-api 2.1</td>
</tr>
<tr>
<td></td>
<td>1.1022.1 Available under license</td>
</tr>
<tr>
<td>1.1023</td>
<td>nocache 1.0.1 1.0.1</td>
</tr>
<tr>
<td></td>
<td>1.1023.1 Available under license</td>
</tr>
<tr>
<td>1.1024</td>
<td>path-parse 1.0.6 1.0.6</td>
</tr>
<tr>
<td></td>
<td>1.1024.1 Available under license</td>
</tr>
<tr>
<td>1.1025</td>
<td>@babel/plugin-syntax-export-namespace-from 7.8.3</td>
</tr>
<tr>
<td></td>
<td>1.1025.1 Available under license</td>
</tr>
<tr>
<td>1.1026</td>
<td>hive 3.1.2</td>
</tr>
<tr>
<td></td>
<td>1.1026.1 Available under license</td>
</tr>
<tr>
<td></td>
<td>1.1027.1 Available under license</td>
</tr>
<tr>
<td>1.1028</td>
<td>metrics-json 4.1.0</td>
</tr>
<tr>
<td>1.1029</td>
<td>findbugs-annotations 1.3.9-1</td>
</tr>
<tr>
<td></td>
<td>1.1030.1 Available under license</td>
</tr>
<tr>
<td>1.1031</td>
<td>copy-to-clipboard 3.3.1 3.3.1</td>
</tr>
<tr>
<td></td>
<td>1.1031.1 Available under license</td>
</tr>
<tr>
<td>1.1032</td>
<td>commons-validator 1.6</td>
</tr>
<tr>
<td></td>
<td>1.1032.1 Available under license</td>
</tr>
<tr>
<td>1.1033</td>
<td>jackson-databind 2.4.0</td>
</tr>
<tr>
<td></td>
<td>1.1033.1 Available under license</td>
</tr>
<tr>
<td>1.1034</td>
<td>reap2 1.0.1 1.0.1</td>
</tr>
<tr>
<td></td>
<td>1.1034.1 Available under license</td>
</tr>
<tr>
<td>1.1035</td>
<td>crypto-random-string 2.0.0</td>
</tr>
<tr>
<td></td>
<td>1.1035.1 Available under license</td>
</tr>
<tr>
<td>1.1036</td>
<td>argparse 1.0.10 1.0.10</td>
</tr>
<tr>
<td></td>
<td>1.1036.1 Available under license</td>
</tr>
<tr>
<td>1.1037</td>
<td>@jsreport/mingo 2.4.1 2.4.1</td>
</tr>
<tr>
<td></td>
<td>1.1037.1 Available under license</td>
</tr>
<tr>
<td>1.1038</td>
<td>commander 4.1.1 4.1.1</td>
</tr>
<tr>
<td></td>
<td>1.1038.1 Available under license</td>
</tr>
<tr>
<td>1.1039</td>
<td>javolution 5.5.1</td>
</tr>
<tr>
<td></td>
<td>1.1039.1 Available under license</td>
</tr>
<tr>
<td>1.1040</td>
<td>isobject 2.1.0 2.1.0</td>
</tr>
<tr>
<td></td>
<td>1.1040.1 Available under license</td>
</tr>
</tbody>
</table>
1.1041 compression 1.6.2 1.6.2
1.1041.1 Available under license

1.1042 to-readable-stream 1.0.0
1.1042.1 Available under license

1.1043 is-stream 1.1.0 1.1.0
1.1043.1 Available under license

1.1044 antlr 3.1.3
1.1044.1 Available under license

1.1045 has 1.0.3 1.0.3
1.1045.1 Available under license

1.1046 ratelimitj-core 0.4.0
1.1046.1 Available under license

1.1047 lz4-java 1.7.1
1.1047.1 Available under license

1.1048 bluebird 3.5.1 3.5.1
1.1048.1 Available under license

1.1049 mime-pull 1.9.3
1.1050 drill-opentsdb-storage 1.19.0
1.1050.1 Available under license

1.1051 okio 2.4.3
1.1051.1 Available under license

1.1052 babel-plugin-syntax-object-rest-spread 6.13.0
1.1052.1 Available under license

1.1053 socket.io-client 1.7.4 1.7.4
1.1053.1 Available under license

1.1054 jsreport-html-to-xlsx 2.8.3 2.8.3
1.1054.1 Available under license

1.1055 babel-traverse 6.26.0
1.1055.1 Available under license

1.1056 de.huxhorn.sulky.codec 0.9.17
1.1056.1 Available under license

1.1057 jersey-media-jaxb 2.32
1.1057.1 Available under license

1.1058 yargs-parser 13.1.2 13.1.2
1.1058.1 Available under license

1.1059 vary 1.0.1 1.0.1
1.1059.1 Available under license

1.1060 drill-yarn 1.19.0
1.1060.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>async-limiter</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>jnr-netdb</td>
<td>1.1.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>array-slice</td>
<td>1.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>jsreport-scripts</td>
<td>2.6.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>xmlhttprequest-ssl</td>
<td>1.5.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>tiger-types</td>
<td>1.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>hbase-http</td>
<td>2.4.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>@types/node</td>
<td>12.12.68</td>
<td>Available under license</td>
</tr>
<tr>
<td>amcharts3-dataloader</td>
<td>1.0.19</td>
<td>Available under license</td>
</tr>
<tr>
<td>core-js</td>
<td>2.5.7</td>
<td>Available under license</td>
</tr>
<tr>
<td>bcel</td>
<td>2.7.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>require_optional</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>side-channel</td>
<td>1.0.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>got</td>
<td>9.6.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>string-width</td>
<td>4.2.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>strip-ansi</td>
<td>6.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>jcl-over-slf4j</td>
<td>1.7.26</td>
<td></td>
</tr>
<tr>
<td>normalize-path</td>
<td>2.1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-preset-stage-3</td>
<td>6.24.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/helper-validator-option</td>
<td>7.14.5</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.1081 bootstrap 4.3.1 4.3.1
   1.1081.1 Available under license
1.1082 jsonwebtoken 7.4.3 7.4.3
   1.1082.1 Available under license
1.1083 moment 2.24.0 2.24.0
   1.1083.1 Available under license
1.1084 jersey-entity-filtering 2.25.1
   1.1084.1 Available under license
1.1085 has-symbols 1.0.0
   1.1085.1 Available under license
1.1086 guava 30.0-jre
   1.1086.1 Available under license
1.1087 @babel/helper-wrap-function 7.16.0
   1.1087.1 Available under license
1.1088 jsesc 0.5.0
   1.1088.1 Available under license
1.1089 activation-api 1.2.1
   1.1089.1 Available under license
1.1090 protobuf-java 3.11.0
   1.1090.1 Available under license
1.1091 @babel/preset-env 7.16.4
   1.1091.1 Available under license
1.1092 lodash-omit 4.5.0 4.5.0
   1.1092.1 Available under license
1.1093 commons-codec 1.13
   1.1093.1 Available under license
1.1094 nimbus-jose-jwt 7.9
   1.1094.1 Available under license
1.1095 cups 2.1.3 4ubuntu0.10
   1.1095.1 Available under license
1.1096 jersey-media-multipart 2.25.1
   1.1096.1 Available under license
1.1097 string-width 1.0.2 1.0.2
   1.1097.1 Available under license
1.1098 jsreport-freeze 2.0.0 2.0.0
   1.1098.1 Available under license
1.1099 babel-plugin-polyfill-corejs2 0.3.0
   1.1099.1 Available under license
1.1100 protobuf 2.6.1-1.3
1.1100.1 Available under license

1.1101 jctools-core 2.1.1
1.1101.1 Available under license

1.1102 toidentifier 1.0.0 1.0.0
1.1102.1 Available under license

1.1103 jackson-jaxrs-base 2.10.4
1.1103.1 Available under license

1.1104 hk2-locator 2.6.1
1.1104.1 Available under license

1.1105 react-highlight-words 0.17.0 0.17.0
1.1105.1 Available under license

1.1106 babel-helper-replace-supers 6.24.1 6.24.1
1.1106.1 Available under license

1.1107 is-extendable 0.1.1
1.1107.1 Available under license

1.1108 deepmerge 2.1.0 2.1.0
1.1108.1 Available under license

1.1109 @babel/plugin-syntax-object-rest-spread 7.8.3
1.1109.1 Available under license

1.1110 batik-ext 1.13
1.1110.1 Available under license

1.1111 randexp 0.5.3 0.5.3
1.1111.1 Available under license

1.1112 hive-upgrade-acid 3.1.2
1.1112.1 Available under license

1.1113 which-module 2.0.0 2.0.0
1.1113.1 Available under license

1.1114 avro 1.9.1
1.1114.1 Available under license

1.1115 hk2-api 2.2.0-b21
1.1115.1 Available under license

1.1116 fast-deep-equal 1.1.0 1.1.0
1.1116.1 Available under license

1.1117 atk 2.18.0 1
1.1117.1 Available under license

1.1118 @babel/template 7.10.4
1.1118.1 Available under license

1.1119 nan 2.14.2 2.14.2
1.1119.1 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Description</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.14.5</td>
<td>tweetnacl</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.0.1</td>
<td>json5</td>
<td>Available under license</td>
</tr>
<tr>
<td>2.0.2</td>
<td>define-property</td>
<td>Available under license</td>
</tr>
<tr>
<td>0.1.7</td>
<td>has-binary</td>
<td>Available under license</td>
</tr>
<tr>
<td>3.3.0</td>
<td>lie</td>
<td>Available under license</td>
</tr>
<tr>
<td>3.2.4</td>
<td>cglib</td>
<td>Available under license</td>
</tr>
<tr>
<td>2.25.1</td>
<td>jersey-media-jaxb</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1.6</td>
<td>bson</td>
<td>Available under license</td>
</tr>
<tr>
<td>2.10.5</td>
<td>jackson-annotations</td>
<td>Available under license</td>
</tr>
<tr>
<td>3.2.2</td>
<td>hadoop-mapreduce-client-common</td>
<td>Available under license</td>
</tr>
<tr>
<td>7.15.7</td>
<td>@babel/helper-validator-identifier</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.19.0</td>
<td>drill-jdbc</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.14.1</td>
<td>dashdash</td>
<td>Available under license</td>
</tr>
<tr>
<td>6.24.1</td>
<td>babel-preset-stage-2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.3.0</td>
<td>object.pick</td>
<td>Available under license</td>
</tr>
<tr>
<td>0.0.1</td>
<td>concat-map</td>
<td>Available under license</td>
</tr>
<tr>
<td>6.2.1</td>
<td>ws</td>
<td>Available under license</td>
</tr>
<tr>
<td>15.1.0</td>
<td>nyc</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.3.1</td>
<td>klaw</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.0.0</td>
<td>fs.realpath</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.1139.1 Available under license

1.1140 **etag** 1.8.1 1.8.1
1.1140.1 Available under license

1.1141 **esprima** 2.7.3
1.1141.1 Available under license

1.1142 **jsonpolicy** 4.0.1 4.0.1
1.1142.1 Available under license

1.1143 **bunyan** 1.8.1 1.8.1
1.1143.1 Available under license

1.1144 **@babel/plugin-transform-object-super** 7.16.0
1.1144.1 Available under license

1.1145 **path-to-regexp** 2.1.0 2.1.0
1.1145.1 Available under license

1.1146.1 Available under license

1.1147 **jnr-ffi** 2.1.5
1.1147.1 Available under license

1.1148 **drill-druid-storage** 1.19.0
1.1148.1 Available under license

1.1149 **cli-cursor** 2.1.0 2.1.0
1.1149.1 Available under license

1.1150 **string-prototype-codepointat** 0.2.1 0.2.1
1.1150.1 Available under license

1.1151 **is-fullwidth-code-point** 2.0.0 2.0.0
1.1151.1 Available under license

1.1152 **schema-utils** 0.4.7 0.4.7
1.1152.1 Available under license

1.1153 **zjsonpatch** 0.3.6
1.1153.1 Available under license

1.1154 **pip** 9.0.1
1.1154.1 Available under license

1.1155 **babel-plugin-transform-es2015-literals** 6.22.0 6.22.0
1.1155.1 Available under license

1.1156 **eclipse-egit-github-core** 2.1.5
1.1156.1 Available under license

1.1157 **jwa** 1.4.1 1.4.1
1.1157.1 Available under license

1.1158 **minlog** 1.3.0
1.1158.1 Available under license

1.1159 **commons-lang3** 3.8.1
1.1159.1 Available under license

1.1160 npm-run-path 2.0.2 2.0.2
   1.1160.1 Available under license

1.1161 babel-plugin-transform-es2015-destructuring 6.23.0 6.23.0
   1.1161.1 Available under license

1.1162 binary 0.3.0 0.3.0
   1.1162.1 Available under license

1.1163 avatica-core 1.17.0
   1.1163.1 Available under license

1.1164 groovy 3.0.8
   1.1164.1 Available under license

1.1165 babel-cli 6.26.0
   1.1165.1 Available under license

1.1166 babel-plugin-transform-es2015-typeof-symbol 6.23.0
   1.1166.1 Available under license

1.1167 verror 1.10.0 1.10.0
   1.1167.1 Available under license

1.1168 datanucleus-api-jdo 4.2.4
   1.1168.1 Available under license

1.1169 jersey-common 2.5.1
   1.1169.1 Available under license

1.1170 batik-svgrasterizer 1.13
   1.1170.1 Available under license

1.1171 hive-shims 3.1.2
   1.1171.1 Available under license

1.1172 keypress 0.1.0 0.1.0
   1.1172.1 Available under license

1.1173 hadoop-yarn-client 3.2.2
   1.1173.1 Available under license

1.1174 isstream 0.1.2 0.1.2
   1.1174.1 Available under license

1.1175 caseless 0.12.0 0.12.0
   1.1175.1 Available under license

1.1176 loader-utils 1.4.0 1.4.0
   1.1176.1 Available under license

1.1177 isarray 1.0.0 1.0.0
   1.1177.1 Available under license

1.1178 color-convert 2.0.1
   1.1178.1 Available under license
1.1179 scylla-driver-core 3.10.2-scylla-1
  1.1179.1 Available under license
1.1180 kerb-util 1.0.0
  1.1180.1 Available under license
1.1181 resourcelocator 1.0.3
  1.1181.1 Available under license
1.1182 hk2-utils 2.5.0-b32
  1.1182.1 Available under license
1.1183 murmur 1.0.0
  1.1183.1 Available under license
1.1184 httpdlog-parser 5.7
  1.1184.1 Available under license
1.1185 unique-string 2.0.0
  1.1185.1 Available under license
1.1186 define-property 0.2.5
  1.1186.1 Available under license
1.1187 drill-format-image 1.19.0
  1.1187.1 Available under license
1.1188 @babel/code-frame 7.0.0
  1.1188.1 Available under license
1.1189 joi 6.10.1 6.10.1
  1.1189.1 Available under license
1.1190 parseurl 1.3.2 1.3.2
  1.1190.1 Available under license
1.1191 hsql-db 1.8.0.10
  1.1191.1 Available under license
1.1192 httpclient5 5.0.3
  1.1192.1 Available under license
1.1193 pkg-dir 2.0.0
  1.1193.1 Available under license
1.1194 jline 2.14.6
  1.1194.1 Available under license
1.1195 parquet-avro 1.12.0
  1.1195.1 Available under license
1.1196 dtrace-provider 0.6.0 0.6.0
  1.1196.1 Available under license
1.1197 map-stream 0.1.0 0.1.0
  1.1197.1 Available under license
1.1198 cross-spawn 6.0.5 6.0.5
1.1198.1 Available under license
1.1199 metrics-jvm 4.1.0
1.1200 commons-digester 1.8.1
  1.1200.1 Available under license
1.1201 ant-antlr 1.10.9
  1.1201.1 Available under license
1.1202 has-values 0.1.4 0.1.4
  1.1202.1 Available under license
1.1203 calcite-linq4j 1.21.0-drill-r2
  1.1203.1 Available under license
1.1204 react-is 16.12.0 16.12.0
  1.1204.1 Available under license
1.1205 json-schema-traverse 0.4.1 0.4.1
  1.1205.1 Available under license
1.1206 jline-terminal-jna 3.12.1
  1.1206.1 Available under license
1.1207 groovy-swing 3.0.8
  1.1207.1 Available under license
1.1208 pdfbox 2.0.22
  1.1208.1 Available under license
1.1209 bops 0.1.1 0.1.1
  1.1209.1 Available under license
1.1210 jsonpath-plus 3.0.0 3.0.0
  1.1210.1 Available under license
1.1211 xpp 1.1.4c
  1.1211.1 Notifications
  1.1211.2 Available under license
1.1212 asm 5.0.3
  1.1212.1 Available under license
1.1213 big.js 5.2.2 5.2.2
  1.1213.1 Available under license
1.1214 type-fest 0.20.2
  1.1214.1 Available under license
1.1215 arraybuffer.slice 0.0.7 0.0.7
  1.1215.1 Available under license
1.1216 json-schema-ref-parser 6.1.0 6.1.0
  1.1216.1 Available under license
1.1217 async 0.9.0 0.9.0
  1.1217.1 Available under license
1.1218 babel-preset-env 1.7.0 1.7.0
   1.1218.1 Available under license
1.1219 component-emitter 1.3.0 1.3.0
   1.1219.1 Available under license
1.1220 curator-framework 5.1.0
   1.1220.1 Available under license
1.1221 hbase-asyncfs 2.4.2
   1.1221.1 Available under license
1.1222 jetty-security 9.4.41.v20210516
   1.1222.1 Available under license
1.1223 groovy-xml 3.0.8
   1.1223.1 Available under license
1.1224 @babel/parser 7.11.5
   1.1224.1 Available under license
1.1225 expand-brackets 2.1.4 2.1.4
   1.1225.1 Available under license
1.1226 metrics-health-checks 4.0.2
1.1227 tomcat-annotations-api 8.5.46
   1.1227.1 Available under license
1.1228 hadoop-client 3.2.2
   1.1228.1 Available under license
1.1229 babel-helper-hoist-variables 6.24.1 6.24.1
   1.1229.1 Available under license
1.1230 is-descriptor 0.1.6 0.1.6
   1.1230.1 Available under license
1.1231 picocli 4.5.2
   1.1231.1 Available under license
1.1232 protobuf-java 2.5.32
   1.1232.1 Available under license
1.1233 libxcb 1.11.1 1ubuntu1
   1.1233.1 Available under license
1.1234 kerby-config 1.0.0
   1.1234.1 Available under license
1.1235 shelljs 0.3.0 0.3.0
   1.1235.1 Available under license
1.1236 has-own-deep 1.1.0 1.1.0
   1.1236.1 Available under license
1.1237 batik-rasterizer-ext 1.13
   1.1237.1 Available under license
<table>
<thead>
<tr>
<th>1.1238</th>
<th>uri-js 4.4.1 4.4.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1238.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1239</td>
<td>diff2html 2.5.0 2.5.0</td>
</tr>
<tr>
<td>1.1239.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1240</td>
<td>ignite-core 2.11.0</td>
</tr>
<tr>
<td>1.1240.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1241</td>
<td>convert-source-map 1.7.0</td>
</tr>
<tr>
<td>1.1241.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1242</td>
<td>batik 1.13</td>
</tr>
<tr>
<td>1.1242.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1243</td>
<td>is-callable 1.2.2</td>
</tr>
<tr>
<td>1.1243.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1244</td>
<td>emojis-list 2.1.0 2.1.0</td>
</tr>
<tr>
<td>1.1244.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1245</td>
<td>uri-js 4.2.2 4.2.2</td>
</tr>
<tr>
<td>1.1245.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1246</td>
<td>buffer-shims 1.0.0 1.0.0</td>
</tr>
<tr>
<td>1.1246.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1247</td>
<td>activation 1.1</td>
</tr>
<tr>
<td>1.1247.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1248</td>
<td>activation-api 1.2.0</td>
</tr>
<tr>
<td>1.1248.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1249</td>
<td>babel-plugin-transform-strict-mode 6.24.1</td>
</tr>
<tr>
<td>1.1249.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1250</td>
<td>gson 2.8.5</td>
</tr>
<tr>
<td>1.1250.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1251</td>
<td>split 0.3.3 0.3.3</td>
</tr>
<tr>
<td>1.1251.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1252</td>
<td>@babel/plugin-syntax-numeric-separator 7.10.4</td>
</tr>
<tr>
<td>1.1252.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1253</td>
<td>javaparser-core 3.18.0</td>
</tr>
<tr>
<td>1.1253.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1254</td>
<td>amqplib-tutorials 0.0.1 0.0.1</td>
</tr>
<tr>
<td>1.1254.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1255</td>
<td>source-map 0.2.0</td>
</tr>
<tr>
<td>1.1255.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1256</td>
<td>statuses 1.4.0 1.4.0</td>
</tr>
<tr>
<td>1.1256.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1257</td>
<td>avro 1.7.7</td>
</tr>
<tr>
<td>Version</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>1.1257.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1258</td>
<td>cliui 5.0.0</td>
</tr>
<tr>
<td>1.1259</td>
<td>static-extend 0.1.2 0.1.2</td>
</tr>
<tr>
<td>1.1260</td>
<td>loader-utils 1.2.3 1.2.3</td>
</tr>
<tr>
<td>1.1261</td>
<td>regenerator-runtime 0.13.7 0.13.7</td>
</tr>
<tr>
<td>1.1262</td>
<td>batik-constants 1.13</td>
</tr>
<tr>
<td>1.1263</td>
<td>lowercase-keys 1.0.1</td>
</tr>
<tr>
<td>1.1264</td>
<td>snapdragon-util 3.0.1 3.0.1</td>
</tr>
<tr>
<td>1.1265</td>
<td>negotiator 0.6.1 0.6.1</td>
</tr>
<tr>
<td>1.1266</td>
<td>lodash.defaults 4.2.0 4.2.0</td>
</tr>
<tr>
<td>1.1267</td>
<td>to-regex-range 5.0.1 5.0.1</td>
</tr>
<tr>
<td>1.1268</td>
<td>node-extend-without-arrays 1.1.6 1.1.6</td>
</tr>
<tr>
<td>1.1269</td>
<td>lodash-some 4.6.0 4.6.0</td>
</tr>
<tr>
<td>1.1270</td>
<td>jsreport-fs-store 2.9.1 2.9.1</td>
</tr>
<tr>
<td>1.1271</td>
<td>is-fullwidth-code-point 2.0.0</td>
</tr>
<tr>
<td>1.1272</td>
<td>readdirp 3.2.0</td>
</tr>
<tr>
<td>1.1273</td>
<td>jsp 2.1</td>
</tr>
<tr>
<td>1.1274</td>
<td>babel-plugin-syntax-dynamic-import 6.18.0</td>
</tr>
<tr>
<td>1.1275</td>
<td>formidable 1.2.2 1.2.2</td>
</tr>
<tr>
<td>1.1276</td>
<td>protobuf-java 2.5.0</td>
</tr>
</tbody>
</table>
1.1277 checker-qual 2.10.0
  1.1277.1 Available under license
1.1278 hbase-hadoop-compat 2.0.0-alpha4
  1.1278.1 Available under license
1.1279 glob-parent 5.1.1
  1.1279.1 Available under license
1.1280 foreground-child 2.0.0
  1.1280.1 Available under license
1.1281 commons-text 1.6
  1.1281.1 Available under license
1.1282 hive-serde 3.1.2
  1.1282.1 Available under license
1.1283 zero-allocation-hashing 0.9
  1.1283.1 Available under license
1.1284 accepts 1.3.3 1.3.3
  1.1284.1 Available under license
1.1285 jsreport-client 1.2.0 1.2.0
  1.1285.1 Available under license
1.1286 wayland 1.12.0-1~ubuntu16.04.3
  1.1286.1 Available under license
1.1287 commons-lang3 2.4
  1.1287.1 Available under license
1.1288 babel-plugin-syntax-decorators 6.13.0
  1.1288.1 Available under license
1.1289 commons-logging 1.1.1
  1.1289.1 Available under license
1.1290 hadoop-hdfs 3.2.2
  1.1290.1 Available under license
1.1291 file-type 11.1.0 11.1.0
  1.1291.1 Available under license
1.1292 type-fest 0.8.1
  1.1292.1 Available under license
1.1293 reselect 3.0.1 3.0.1
  1.1293.1 Available under license
1.1294 moment-timezone 0.5.31 0.5.31
  1.1294.1 Available under license
1.1295 drill-storage-splunk 1.19.0
  1.1295.1 Available under license
1.1296 json-smart 2.3
1.1296.1 Available under license
1.1297 set-blocking 2.0.0 2.0.0
  1.1297.1 Available under license
1.1298 duplexify 3.7.1 3.7.1
  1.1298.1 Available under license
1.1299 lodash-es 4.17.15 4.17.15
  1.1299.1 Available under license
1.1300 jackson 2.12.3
  1.1300.1 Available under license
1.1301 aws-java-sdk-kms 1.11.375
  1.1301.1 Available under license
1.1302 slash 3.0.0 3.0.0
  1.1302.1 Available under license
1.1303 registry-auth-token 4.2.1
  1.1303.1 Available under license
1.1304 libbsd 0.8.2 1ubuntu0.1
  1.1304.1 Available under license
1.1305 c3p0 0.9.5.4
  1.1305.1 Available under license
1.1306 hive-llap-client 3.1.2
  1.1306.1 Available under license
1.1307 mongodb-migrations 1.0.1 1.0.1
  1.1307.1 Available under license
1.1308 call-bind 1.0.2
  1.1308.1 Available under license
1.1309 word-wrap 1.2.3
  1.1309.1 Available under license
1.1310 batik-i18n 1.13
  1.1310.1 Available under license
1.1311 xdg-basedir 4.0.0
  1.1311.1 Available under license
1.1312 which 1.3.1 1.3.1
  1.1312.1 Available under license
1.1313 my-scope-package-a 0.0.0
  1.1313.1 Available under license
1.1314 trim-right 1.0.1
  1.1314.1 Available under license
1.1315 ajv 6.12.6 6.12.6
  1.1315.1 Available under license
1.1316 core-util-is 1.0.2 1.0.2
  1.1316.1 Available under license
1.1317 datanucleus-core 4.1.17
  1.1317.1 Available under license
1.1318 negotiator 0.6.2 0.6.2
  1.1318.1 Available under license
1.1319 is-stream 2.0.0
  1.1319.1 Available under license
1.1320 glob-parent 5.1.1 5.1.1
  1.1320.1 Available under license
1.1321 base64id 2.0.0 2.0.0
  1.1321.1 Available under license
1.1322 ipaddr.js 1.9.1 1.9.1
  1.1322.1 Available under license
1.1323 readdirp 3.3.0 3.3.0
  1.1323.1 Available under license
1.1324 electron-to-chromium 1.3.904
  1.1324.1 Available under license
1.1325 lodash-pick 4.4.0 4.4.0
  1.1325.1 Available under license
1.1326 elfutils 0.165 3ubuntu1.2
  1.1326.1 Available under license
1.1327 jetty 9.4.34.v20201102
  1.1327.1 Available under license
1.1328 onetime 1.1.0
  1.1328.1 Available under license
1.1329 p-reduce 2.1.0 2.1.0
  1.1329.1 Available under license
1.1330 is-fullwidth-code-point 1.0.0 1.0.0
  1.1330.1 Available under license
1.1331 mixin-deep 1.3.2 1.3.2
  1.1331.1 Available under license
1.1332 diff 3.5.0 3.5.0
  1.1332.1 Available under license
1.1333 mime-db 1.44.0 1.44.0
  1.1333.1 Available under license
1.1334 babel-plugin-syntax-async-functions 6.13.0 6.13.0
  1.1334.1 Available under license
1.1335 unpipe 1.0.0 1.0.0
1.1335.1 Available under license

1.1336 \texttt{readable-stream} 3.6.0 3.6.0
1.1336.1 Available under license

1.1337 \texttt{moment} 2.22.2 2.22.2
1.1337.1 Available under license

1.1338 \texttt{guava} 30.1-jre
1.1338.1 Available under license

1.1339 \texttt{request} 2.87.0 2.87.0
1.1339.1 Available under license

1.1340 \texttt{janino} 3.0.11

1.1341 \texttt{ansi-styles} 4.2.1
1.1341.1 Available under license

1.1342 \texttt{find-up} 3.0.0
1.1342.1 Available under license

1.1343 \texttt{zookeeper} 3.5.7
1.1343.1 Available under license

1.1344 \texttt{arr-diff} 4.0.0 4.0.0
1.1344.1 Available under license

1.1345 \texttt{jackson-databind} 2.12.1
1.1345.1 Available under license

1.1346 \texttt{cliui} 7.0.4 7.0.4
1.1346.1 Available under license

1.1347 \texttt{eslint-plugin-import} 2.25.3
1.1347.1 Available under license

1.1348 \texttt{jcip-annotation} 1.0.1
1.1348.1 Available under license

1.1349 \texttt{mimic-fn} 1.2.0 1.2.0
1.1349.1 Available under license

1.1350 \texttt{is-number-object} 1.0.4
1.1350.1 Available under license

1.1351 \texttt{q} 0.9.7 0.9.7
1.1351.1 Available under license

1.1352 \texttt{jakarta-validation-api} 2.0.2
1.1352.1 Available under license

1.1353 \texttt{graphite} 1.3.10-0ubuntu0.16.04.1
1.1353.1 Available under license

1.1354 \texttt{failureaccess} 1.0.1

1.1355 \texttt{orc-shims} 1.5.1
1.1355.1 Available under license
<table>
<thead>
<tr>
<th>1.1356 methods 1.1.2</th>
<th>1.1.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1356.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1357 api-spec-converter 2.12.0</th>
<th>2.12.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1357.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1358 apache-log4j 2.14.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1358.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1359 asm-tree 7.3.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1359.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1360 libpng 1.6.35</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1360.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1361 @babel/plugin-transform-parameters 7.16.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1361.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1362 foodmart-data-json 0.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1362.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1363 metrics-jvm 4.0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1364.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1365 bluebird 3.7.2</th>
<th>3.7.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1365.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1366 require-uncached 1.0.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1366.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1367 xml-beans 4.0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1367.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1368 object-component 0.0.3</th>
<th>0.0.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1368.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1369 jetty-security 9.4.34.v20201102</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1369.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1370 shebang-command 1.2.0</th>
<th>1.2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1370.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1371 table 3.8.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1371.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1372 browserslist 3.2.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1372.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1373 latencyutils 2.0.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1373.1 Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1374 is-accessor-descriptor 1.0.0</th>
<th>1.0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1374.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1375 diff-match-patch 1.0.5</th>
<th>1.0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1375.1 Available under license</td>
<td></td>
</tr>
</tbody>
</table>

| 1.1376 javax-ws-rs-api 2.0.1 |
1.1376.1 Available under license
1.1377 jersey-container-servlet-core 2.25.1
  1.1377.1 Available under license
1.1378 drange 1.1.1 1.1.1
  1.1378.1 Available under license
1.1379 jnr-constants 0.9.0
1.1380 parse-passwd 1.0.0 1.0.0
  1.1380.1 Available under license
1.1381 async 1.5.2
  1.1381.1 Available under license
1.1382 netty-tcnative 2.0.1.First
  1.1382.1 Available under license
1.1383 swagger-methods 1.0.8 1.0.8
  1.1383.1 Available under license
1.1384 tar-stream 1.6.2 1.6.2
  1.1384.1 Available under license
1.1385 jsr305 3.0.2
  1.1385.1 Available under license
1.1386 babel-plugin-transform-property-literals 7.16.0
  1.1386.1 Available under license
1.1387 static-eval 2.0.2 2.0.2
  1.1387.1 Available under license
1.1388 core-js 3.19.1
  1.1388.1 Available under license
1.1389 msexcel-builder-extended 0.0.8 0.0.8
  1.1389.1 Available under license
1.1390 babel-plugin-syntax-trailing-function-commas 6.22.0 6.22.0
  1.1390.1 Available under license
1.1391 influx 5.0.7 5.0.7
  1.1391.1 Available under license
1.1392 decamelize 1.2.0 1.2.0
  1.1392.1 Available under license
1.1393 output-file-sync 1.1.2
  1.1393.1 Available under license
1.1394 streamsearch 0.1.2 0.1.2
  1.1394.1 Available under license
1.1395 stream-to-array 2.3.0 2.3.0
  1.1395.1 Available under license
1.1396 buffer-more-ints 1.0.0 1.0.0
1.1396.1 Available under license

1.1397 find-cache-dir 2.1.0
   1.1397.1 Available under license

1.1398 jtoml 1.0.0
   1.1398.1 Available under license

1.1399 jakarta-ws-rs-api 2.1.6
   1.1399.1 Available under license

1.1400 p-cancelable 1.1.0
   1.1400.1 Available under license

1.1401 has-yarn 2.1.0
   1.1401.1 Available under license

1.1402 szmarczak-http-timer 1.1.2
   1.1402.1 Available under license

1.1403 send 0.16.2 0.16.2
   1.1403.1 Available under license

1.1404 es6-map 0.1.5
   1.1404.1 Available under license

1.1405 setprototypeof 1.0.3 1.0.3
   1.1405.1 Available under license

1.1406 fromentries 1.3.2
   1.1406.1 Available under license

1.1407 forever-agent 0.6.1 0.6.1
   1.1407.1 Available under license

1.1408 babel-plugin-transform-es2015-arrow-functions 6.22.0
   1.1408.1 Available under license

1.1409 jacoco-agent 0.7.9
   1.1409.1 Available under license

1.1410 babel-plugin-transform-es2015-block-scoped-functions 6.22.0 6.22.0
   1.1410.1 Available under license

1.1411 wrap-ansi 5.1.0 5.1.0
   1.1411.1 Available under license

1.1412 xmlbuilder 9.0.4 9.0.4
   1.1412.1 Available under license

1.1413 object-visit 1.0.1 1.0.1
   1.1413.1 Available under license

1.1414 jetty-webapp 9.4.41.v20210516
   1.1414.1 Available under license

1.1415 commons-io 2.11.0
   1.1415.1 Available under license
1.1416 oro 2.0.8
   1.1416.1 Available under license

1.1417 qt 5.7.0
   1.1417.1 Available under license

1.1418 parquet-format 2.3.0-incubating
   1.1418.1 Available under license

1.1419 fast-deep-equal 3.1.3 3.1.3
   1.1419.1 Available under license

1.1420 @babel/plugin-transform-arrow-functions 7.16.0
   1.1420.1 Available under license

1.1421 emoji-regex 8.0.0 8.0.0
   1.1421.1 Available under license

1.1422 commons-text 1.9
   1.1422.1 Available under license

1.1423 ci-info 2.0.0
   1.1423.1 Available under license

1.1424 ignite-shmem 1.0.0
   1.1424.1 Available under license

1.1425 dicer 0.2.5 0.2.5
   1.1425.1 Available under license

1.1426 to-regex-range 5.0.1
   1.1426.1 Available under license

1.1427 is-number 3.0.0 3.0.0
   1.1427.1 Available under license

1.1428 path-exists 3.0.0 3.0.0
   1.1428.1 Available under license

1.1429 number-is-nan 1.0.1 1.0.1
   1.1429.1 Available under license

1.1430 httpcomponents-client 4.5.6
   1.1430.1 Available under license

1.1431 object-assign 4.1.0 4.1.0
   1.1431.1 Available under license

1.1432 resolve 1.18.1 1.18.1
   1.1432.1 Available under license

1.1433 avro-mapred 1.9.1
   1.1433.1 Available under license

1.1434 lodash.assign 4.2.0
   1.1434.1 Available under license

1.1435 mime 2.4.6 2.4.6
1.1435.1 Available under license

1.1436 @babel/cli 7.16.0
   1.1436.1 Available under license

1.1437 commons-logging 1.2
   1.1437.1 Available under license

   1.1438.1 Available under license

1.1439 glob 6.0.4 6.0.4
   1.1439.1 Available under license

1.1440 fill-range 4.0.0 4.0.0
   1.1440.1 Available under license

1.1441 babel-plugin-transform-es2015-sticky-regex 6.24.1
   1.1441.1 Available under license

1.1442 de.huxhorn.lilith.logback.appender.multiplex-classic 0.9.44
   1.1442.1 Available under license

1.1443 jakarta xml bind api 2.3.2
   1.1443.1 Available under license

1.1444 assert-plus 1.0.0 1.0.0
   1.1444.1 Available under license

1.1445 object.assign 4.1.1
   1.1445.1 Available under license

1.1446 yargs-parser 18.1.3
   1.1446.1 Available under license

1.1447 prepend-http 1.0.4 1.0.4
   1.1447.1 Available under license

1.1448 fastutil 6.5.7
   1.1448.1 Available under license

1.1449 is-extglob 2.1.1 2.1.1
   1.1449.1 Available under license

1.1450 uuid 3.3.2 3.3.2
   1.1450.1 Available under license

1.1451 lm-sensors 3.4.0-2
   1.1451.1 Available under license

1.1452 readable-stream 2.2.7 2.2.7
   1.1452.1 Available under license

1.1453 jetty-webapp 9.4.34.v20201102
   1.1453.1 Available under license

1.1454 babel-helper-regex 6.26.0
   1.1454.1 Available under license
<table>
<thead>
<tr>
<th></th>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1455</td>
<td>thrift</td>
<td>0.14.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1456</td>
<td>junit</td>
<td>4.10</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1457</td>
<td>engine.io-client</td>
<td>1.8.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1458</td>
<td>pako</td>
<td>1.0.11</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1459</td>
<td>lodash</td>
<td>4.17.20</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1460</td>
<td>babel-helper-replace-supers</td>
<td>6.24.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1461</td>
<td>junit-jupiter-api</td>
<td>5.7.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1462</td>
<td>aspectj-tools</td>
<td>1.8.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1463</td>
<td>de.huxhorn.lilith.logback.appender.multiplex-core</td>
<td>0.9.44</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1464</td>
<td>domhandler</td>
<td>2.4.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1465</td>
<td>qs</td>
<td>6.7.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1466</td>
<td>guava</td>
<td>30.0-jre</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1467</td>
<td>jetty-webapp</td>
<td>9.3.27.v20190418</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1468</td>
<td>os-homedir</td>
<td>1.0.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1469</td>
<td>duplexer2</td>
<td>0.1.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1470</td>
<td>object-keys</td>
<td>1.1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1471</td>
<td>immediate</td>
<td>3.0.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1472</td>
<td>kite-hadoop-compatibility</td>
<td>1.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1473</td>
<td>drill-rdbms-metastore</td>
<td>1.19.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1474</td>
<td>libthai</td>
<td>0.1.24</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.1475</td>
<td>cglib</td>
<td>2.2.2</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.1475.1 Available under license
1.1476 **any.promise** 1.3.0 1.3.0
   1.1476.1 Available under license
1.1477 **array-flatten** 1.1.1 1.1.1
   1.1477.1 Available under license
1.1478 **pify** 4.0.1 4.0.1
   1.1478.1 Available under license
1.1479 **parsejson** 0.0.3 0.0.3
   1.1479.1 Available under license
1.1480 **code-point-at** 1.1.0 1.1.0
   1.1480.1 Available under license
1.1481 **buffer-fill** 1.0.0 1.0.0
   1.1481.1 Available under license
1.1482 **buffers** 0.1.1 0.1.1
   1.1482.1 Available under license
1.1483 **micromatch** 2.3.11
   1.1483.1 Available under license
1.1484 **to-array** 0.1.4 0.1.4
   1.1484.1 Available under license
1.1485 **is-descriptor** 1.0.2 1.0.2
   1.1485.1 Available under license
1.1486 **batik-anim** 1.13
   1.1486.1 Available under license
1.1487 **babel-helper-define-map** 6.26.0
   1.1487.1 Available under license
1.1488 **propagate** 2.0.1
   1.1488.1 Available under license
1.1489 **iconv-lite** 0.4.24 0.4.24
   1.1489.1 Available under license
1.1490 **@babel/plugin-syntax-async-generators** 7.8.4
   1.1490.1 Available under license
1.1491 **kerb-server** 1.0.0
   1.1491.1 Available under license
1.1492 **content-security-policy-builder** 1.0.0 1.0.0
   1.1492.1 Available under license
1.1493 **run-async** 0.1.0
   1.1493.1 Available under license
1.1494 **serializator** 1.0.2 1.0.2
   1.1494.1 Available under license
1.1495 p-limit 1.3.0
   1.1495.1 Available under license
1.1496 pstree-remy 1.1.8
   1.1496.1 Available under license
1.1497 write-file-atomic 3.0.3
   1.1497.1 Available under license
1.1498 jsreport-base 2.0.2 2.0.2
   1.1498.1 Available under license
1.1499 config 1.3.3
   1.1499.1 Available under license
1.1500 events 2.1.0 2.1.0
   1.1500.1 Available under license
1.1501 kerby-util 1.0.0
   1.1501.1 Available under license
1.1502 hadoop-yarn-common 3.2.2
   1.1502.1 Available under license
1.1503 lazy 1.0.11 1.0.11
   1.1503.1 Available under license
1.1504 regexu-core 4.8.0
   1.1504.1 Available under license
1.1505 p-is-promise 2.1.0 2.1.0
   1.1505.1 Available under license
1.1506 httpcomponents-core 4.4.13
   1.1506.1 Available under license
1.1507 babel-plugin-check-es2015-constants 6.22.0 6.22.0
   1.1507.1 Available under license
1.1508 websocket-servlet 9.3.27.v20190418
1.1509 babel-plugin-transform-regenerator 6.26.0 6.26.0
   1.1509.1 Available under license
1.1510 regex-not 1.0.2 1.0.2
   1.1510.1 Available under license
1.1511 guice 5.0.1
   1.1511.1 Available under license
1.1512 is-resolvable 1.1.0
   1.1512.1 Available under license
1.1513 babel-register 7.16.0
   1.1513.1 Available under license
1.1514 drill-metastore-api 1.19.0
   1.1514.1 Available under license
1.1515 *denque* 1.5.0 1.5.0
   1.1515.1 Available under license
1.1516 *utf-8-validate* 1.2.1 1.2.1
   1.1516.1 Available under license
1.1517 *support-color* 6.0.0
   1.1517.1 Available under license
1.1518 *fast-deep-equal* 2.0.1 2.0.1
   1.1518.1 Available under license
1.1519 *event-emitter* 0.3.5
   1.1519.1 Available under license
1.1520 *jctools-core* 3.1.0
   1.1520.1 Available under license
1.1521 *babel-plugin-syntax-exponentiation-operator* 6.13.0
   1.1521.1 Available under license
1.1522 *asm* 7.3.1
   1.1522.1 Available under license
1.1523 *utils-merge* 1.0.0 1.0.0
   1.1523.1 Available under license
1.1524 *bytes* 3.1.0 3.1.0
   1.1524.1 Available under license
1.1525 *mime-types* 2.1.27 2.1.27
   1.1525.1 Available under license
1.1526 *paranamer* 2.5.6
1.1527 *opentracing-api* 0.31.0
   1.1527.1 Available under license
1.1528 *snappy* 1.0.5
   1.1528.1 Available under license
1.1529 *brace-expansion* 1.1.11 1.1.11
   1.1529.1 Available under license
1.1530 *lodash.isundefined* 3.0.1 3.0.1
   1.1530.1 Available under license
1.1531 *@babel/plugin-syntax-logical-assignment-operators* 7.10.4
   1.1531.1 Available under license
1.1532 *@babel/traverse* 7.11.5
   1.1532.1 Available under license
1.1533 *jasper-compiler* 5.5.12
   1.1533.1 Available under license
1.1534 *safe-buffer* 5.2.1 5.2.1
   1.1534.1 Available under license
1.1535 after 0.8.2 0.8.2
  1.1535.1 Available under license
1.1536 antlr 3.5.2
  1.1536.1 Available under license
1.1537 avatica-core 1.15.0
  1.1537.1 Available under license
1.1538 protobuf-java-util 3.13.0
  1.1538.1 Available under license
1.1539 utile 0.3.0 0.3.0
  1.1539.1 Available under license
1.1540 fs.realpath 1.0.0 1.0.0
  1.1540.1 Available under license
1.1541 amcharts3-responsive 1.0.5 1.0.5
  1.1541.1 Available under license
1.1542 socket.io-parser 2.3.1 2.3.1
  1.1542.1 Available under license
1.1543 slf4j 1.7.2
  1.1543.1 Available under license
1.1544 metrics-core 2.2.0
  1.1544.1 Available under license
1.1545 jersey-media-json-jackson 2.25.1
  1.1545.1 Available under license
1.1546 chalk 0.4.0 0.4.0
  1.1546.1 Available under license
1.1547 on-headers 1.0.1 1.0.1
  1.1547.1 Available under license
1.1548 lodash.isplainobject 4.0.6 4.0.6
  1.1548.1 Available under license
1.1549 jakarta-inject 2.6.1
  1.1549.1 Available under license
1.1550 is-accessor-descriptor 0.1.6
  1.1550.1 Available under license
1.1551 shelljs 0.8.4 0.8.4
  1.1551.1 Available under license
1.1552 query-string 7.0.0 7.0.0
  1.1552.1 Available under license
1.1553 fontconfig 2.11.94 Ubuntu1.1
  1.1553.1 Available under license
1.1554 chokidar 3.3.0
<table>
<thead>
<tr>
<th>Package name</th>
<th>Version</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>nodemon</td>
<td>2.0.15</td>
<td>Available under license</td>
</tr>
<tr>
<td>lodash-set</td>
<td>4.3.2</td>
<td>4.3.2</td>
</tr>
<tr>
<td>slice-ansi</td>
<td>0.0.4</td>
<td>0.0.4</td>
</tr>
<tr>
<td>commons-config</td>
<td>1.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>pdfjs-dist</td>
<td>2.0.489</td>
<td>2.0.489</td>
</tr>
<tr>
<td>babel-plugin-check-es2015-constants</td>
<td>6.22.0</td>
<td>6.22.0</td>
</tr>
<tr>
<td>defer-to-connect</td>
<td>1.1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>jsreport-static-pdf</td>
<td>0.4.0</td>
<td>0.4.0</td>
</tr>
<tr>
<td>buffer-alloc</td>
<td>1.2.0</td>
<td>1.2.0</td>
</tr>
<tr>
<td>nanoid</td>
<td>3.1.12</td>
<td>3.1.12</td>
</tr>
<tr>
<td>normalize-url</td>
<td>4.5.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>graphics2d</td>
<td>0.30</td>
<td>0.30</td>
</tr>
<tr>
<td>wordwrap</td>
<td>1.0.0</td>
<td>1.0.0</td>
</tr>
<tr>
<td>jsreport-assets</td>
<td>1.7.0</td>
<td>1.7.0</td>
</tr>
<tr>
<td>cookie</td>
<td>0.1.5</td>
<td>0.1.5</td>
</tr>
<tr>
<td>ecdsa-sig-formatter</td>
<td>1.0.10</td>
<td>1.0.10</td>
</tr>
<tr>
<td>jackson-jaxrs</td>
<td>2.12.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>ms</td>
<td>2.0.0</td>
<td>2.0.0</td>
</tr>
<tr>
<td>p-defer</td>
<td>1.0.0</td>
<td>1.0.0</td>
</tr>
</tbody>
</table>
Open Source Used In DNAC Platform Arica

1.1574 unc-path-regex 0.1.2 0.1.2
  1.1574.1 Available under license
1.1575 dom 1.0
  1.1575.1 Available under license
1.1576 is-property 1.0.2 1.0.2
  1.1576.1 Available under license
1.1577 path-key 2.0.1 2.0.1
  1.1577.1 Available under license
1.1578 is-weakref 1.0.1
  1.1578.1 Available under license
1.1579 mail 1.5.6
  1.1579.1 Available under license
1.1580 kerb-client 1.0.0
  1.1580.1 Available under license
1.1581 kafka-schema-serializer 6.1.1
  1.1581.1 Available under license
1.1582 @babel/helper-skip-transparent-expression-wrappers 7.16.0
  1.1582.1 Available under license
1.1583 ms 2.1.3 2.1.3
  1.1583.1 Available under license
1.1584 javassist 3.18.2-GA
  1.1584.1 Available under license
1.1585 mocha 7.2.0
  1.1585.1 Available under license
1.1586 reactor-core 3.0.7.RELEASE
  1.1586.1 Available under license
1.1587 serve-favicon 2.5.0 2.5.0
  1.1587.1 Available under license
1.1588 deep-sort-object 1.0.2 1.0.2
  1.1588.1 Available under license
1.1589 parser-core 5.7
  1.1589.1 Available under license
1.1590 compressible 2.0.18 2.0.18
  1.1590.1 Available under license
1.1591 follow-redirects 1.14.5 1.14.5
  1.1591.1 Available under license
1.1592 junit-jupiter-engine 5.7.0
  1.1592.1 Available under license
1.1593 babel-preset-es2015 6.24.1
1.1593.1 Available under license

1.1594 faker 4.1.0 4.1.0
1.1594.1 Available under license

1.1595 array-unique 0.3.2 0.3.2
1.1595.1 Available under license

1.1596 tmp 0.1.0 0.1.0
1.1596.1 Available under license

1.1597 bonecp 0.8.0.RELEASE
1.1597.1 Available under license

1.1598 lodash-reject 4.6.0 4.6.0
1.1598.1 Available under license

1.1599 is-relative 1.0.0 1.0.0
1.1599.1 Available under license

1.1600 cookies 0.7.3 0.7.3
1.1600.1 Available under license

1.1601 to-utf8 0.0.1 0.0.1
1.1601.1 Available under license

1.1602 posix-character-classes 0.1.1 0.1.1
1.1602.1 Available under license

1.1603 commons-math 3.1.1
1.1603.1 Available under license

1.1604 streamifier 0.1.1 0.1.1
1.1604.1 Available under license

1.1605 parquet-hadoop 1.10.0
1.1605.1 Available under license

1.1606 metrics-jmx 4.0.2
1.1607 accepts 1.3.7 1.3.7
1.1607.1 Available under license

1.1608 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava
1.1608.1 Available under license

1.1609 batik-util 1.13
1.1609.1 Available under license

1.1610 mime 1.6.0 1.6.0
1.1610.1 Available under license

1.1611 type-is 1.6.16 1.6.16
1.1611.1 Available under license

1.1612 acorn 5.7.4
1.1612.1 Available under license

1.1613 jsoup 1.9.2
1.1613.1 Available under license
1.1614 extend-shallow 2.0.1
   1.1614.1 Available under license
1.1615 jetty-util 9.3.27.v20190418
1.1616 word-wrap 1.2.3 1.2.3
   1.1616.1 Available under license
1.1617 async 2.6.0 2.6.0
   1.1617.1 Available under license
1.1618 hamcrest-library 1.3
   1.1618.1 Available under license
1.1619 lucene-queryparser 7.4.0
   1.1619.1 Available under license
1.1620 model-mapper 1.1.0
1.1621 kafka-raft 2.8.0
   1.1621.1 Available under license
1.1622 bl 1.2.3 1.2.3
   1.1622.1 Available under license
1.1623 core-js-compat 3.19.1
   1.1623.1 Available under license
1.1624 jersey-container-jetty-http 2.25.1
   1.1624.1 Available under license
1.1625 esutils 2.0.3
   1.1625.1 Available under license
1.1626 glob 7.1.6
   1.1626.1 Available under license
1.1627 tomcat 8.5.46
   1.1627.1 Available under license
1.1628 generate-schema 2.6.0 2.6.0
   1.1628.1 Available under license
1.1629 fd-slicer 1.1.0 1.1.0
   1.1629.1 Available under license
1.1630 expand-brackets 0.1.5
   1.1630.1 Available under license
1.1631 commons-collections 3.2.2
   1.1631.1 Available under license
1.1632 babel-polyfill 6.26.0
1.1633 monorepo-symlink-test 0.0.0
   1.1633.1 Available under license
1.1634 jersey-mvc-freemarker 2.25.1
1.1634.1 Available under license
1.1635 websocket-client 9.3.27.v20190418
1.1636 sindresorhus-is 0.14.0
   1.1636.1 Available under license
1.1637 base16 1.0.0 1.0.0
   1.1637.1 Available under license
1.1638 javassist 3.27.0-GA
   1.1638.1 Available under license
1.1639 drill-protocol 1.19.0
   1.1639.1 Available under license
1.1640 json-path 0.11.2 0.11.2
   1.1640.1 Available under license
1.1641 hbase-prefix-tree 2.0.0-alpha4
   1.1641.1 Available under license
1.1642 commander 2.20.3
   1.1642.1 Available under license
1.1643 jersey 2.32
   1.1643.1 Available under license
1.1644 metrics 0.1.21 0.1.21
   1.1644.1 Available under license
1.1645 batik-squiggle-ext 1.13
   1.1645.1 Available under license
1.1646 sqlite 3.25.3
   1.1646.1 Available under license
1.1647 babel-generator 6.26.1 6.26.1
   1.1647.1 Available under license
1.1648 remark 1.0.0
   1.1648.1 Available under license
1.1649 @babel/helper-plugin-utils 7.14.5
   1.1649.1 Available under license
1.1650 babel-types 6.26.0 6.26.0
   1.1650.1 Available under license
1.1651 js-yaml 3.13.1
   1.1651.1 Available under license
1.1652 lodash-assignin 4.2.0 4.2.0
   1.1652.1 Available under license
1.1653 strip-ansi 4.0.0 4.0.0
   1.1653.1 Available under license
1.1654 cache-base 1.0.1 1.0.1
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>@babel/plugin-transform-block-scoped-functions</td>
<td>7.16.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>groovy-macro</td>
<td>3.0.8</td>
<td>Available under license</td>
</tr>
<tr>
<td>cron-parser</td>
<td>2.7.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>bindings</td>
<td>1.2.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>eventemitter2</td>
<td>0.4.14</td>
<td>Available under license</td>
</tr>
<tr>
<td>shebang-regex</td>
<td>3.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>rimraf</td>
<td>2.4.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>retrofit</td>
<td>2.8.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>ansi-styles</td>
<td>4.3.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>is-shared-array-buffer</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>akka</td>
<td>2.5.29</td>
<td>Available under license</td>
</tr>
<tr>
<td>mime-db</td>
<td>1.47.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>immer</td>
<td>9.0.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>jul-to-slf4j</td>
<td>1.7.26</td>
<td></td>
</tr>
<tr>
<td>jsreport-sample-template</td>
<td>2.6.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>debug</td>
<td>3.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>ncp</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>commons-pool</td>
<td>2.6.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>source-map-support</td>
<td>0.5.21</td>
<td>Available under license</td>
</tr>
<tr>
<td>poi-ooxml</td>
<td>5.0.0</td>
<td></td>
</tr>
</tbody>
</table>
1.1674.1 Available under license

1.1675 ieee754 1.1.13 1.1.13
1.1675.1 Available under license

1.1676 groovy-ant 3.0.8
1.1676.1 Available under license

1.1677 compression 1.7.4 1.7.4
1.1677.1 Available under license

1.1678 frameguard 2.0.0 2.0.0
1.1678.1 Available under license

1.1679 json-simple 1.1.1
1.1680 setprototypeof 1.1.1 1.1.1
1.1680.1 Available under license

1.1681 lowercase-keys 2.0.0
1.1681.1 Available under license

1.1682 libtiff 4.0.6 1ubuntu0.7
1.1682.1 Available under license

1.1683 jetty-client 9.3.27.v20190418
1.1684 multer 1.4.1 1.4.1
1.1684.1 Available under license

1.1685 drill-format-xml 1.19.0
1.1685.1 Available under license

1.1686 readdirp 3.6.0
1.1686.1 Available under license

1.1687 psl 1.8.0 1.8.0
1.1687.1 Available under license

1.1688 libxcursor 1.1.14-1ubuntu0.16.04.2
1.1688.1 Available under license

1.1689 supports-color 2.0.0 2.0.0
1.1689.1 Available under license

1.1690 seamless-immutable 7.1.3 7.1.3
1.1690.1 Available under license

1.1691 shimmer 1.0.0 1.0.0
1.1691.1 Available under license

1.1692 safe-buffer 5.1.2 5.1.2
1.1692.1 Available under license

1.1693 circular-json 0.3.3
1.1693.1 Available under license

1.1694 batik-ttf2svg 1.13
1.1694.1 Available under license
1.1715.1 Available under license
1.1716 **body-parser** 1.18.2 1.18.2
   1.1716.1 Available under license
1.1717 **es-to-primitive** 1.2.1
   1.1717.1 Available under license
1.1718 **avro-ipc-jetty** 1.9.1
   1.1718.1 Available under license
1.1719 **jffi** 1.0.1
   1.1719.1 Available under license
1.1720 **@babel/helper-function-name** 7.10.4
   1.1720.1 Available under license
1.1721 **prop-types** 15.7.2 15.7.2
   1.1721.1 Available under license
1.1722 **metrics-servlets** 4.0.2
1.1723 **kerby-asn1** 1.0.0
   1.1723.1 Available under license
1.1724 **hive-spark-client** 3.1.2
   1.1724.1 Available under license
1.1725 **react-is** 16.13.1 16.13.1
   1.1725.1 Available under license
1.1726 **pump** 3.0.0 3.0.0
   1.1726.1 Available under license
1.1727 **has-ansi** 2.0.0
   1.1727.1 Available under license
1.1728 **aws-java-sdk-core** 1.11.837
   1.1728.1 Available under license
1.1729 **babel-plugin-syntax-async-generators** 6.13.0
   1.1729.1 Available under license
1.1730 **netty-transport-native-unix-common** 4.1.42.Final
   1.1730.1 Available under license
1.1731 **jetty-servlet** 9.4.34.v20201102
   1.1731.1 Available under license
1.1732 **lazystream** 1.0.0 1.0.0
   1.1732.1 Available under license
1.1733 **@babel/types** 7.11.5
   1.1733.1 Available under license
1.1734 **jsreport-text** 2.0.0 2.0.0
   1.1734.1 Available under license
1.1735 **jersey-server** 2.5.1
1.1735.1 Available under license

1.1736 **javax-inject 2.2.0-b21**
   1.1736.1 Available under license

1.1737 **istanbuljs-load-nyc-config 1.1.0**
   1.1737.1 Available under license

1.1738 **react 16.14.0 16.14.0**
   1.1738.1 Available under license

1.1739 **metrics 4.1.0**
   1.1739.1 Available under license

1.1740 **ansi-escapes 1.4.0**
   1.1740.1 Available under license

1.1741 **fbemitter 2.1.1 2.1.1**
   1.1741.1 Available under license

1.1742 **whatwg-fetch 2.0.4 2.0.4**
   1.1742.1 Available under license

1.1743 **stdio 0.2.2 0.2.2**
   1.1743.1 Available under license

1.1744 **encoding 0.1.12 0.1.12**
   1.1744.1 Available under license

1.1745 **jersey-client 2.25.1**
   1.1745.1 Available under license

1.1746 **fast-safe-stringify 2.0.7 2.0.7**
   1.1746.1 Available under license

1.1747 **excel-streaming-reader 3.0.4**
   1.1747.1 Available under license

1.1748 **elasticsearch-rest-client 7.0.1**
   1.1748.1 Available under license

1.1749 **ini 1.3.5 1.3.5**
   1.1749.1 Available under license

1.1750 **swagger-converter 1.5.1 1.5.1**
   1.1750.1 Available under license

1.1751 **pause 0.0.1 0.0.1**
   1.1751.1 Available under license

1.1752 **ajv 6.10.2 6.10.2**
   1.1752.1 Available under license

1.1753 **mongo-java-driver 3.12.4**
   1.1753.1 Available under license

1.1754 **unzipper 0.9.15 0.9.15**
   1.1754.1 Available under license
Open Source Used In DNAC Platform Arica

1.1755 jsreport-authentication 2.6.1 2.6.1
   1.1755.1 Available under license

1.1756 emoji-regex 7.0.3
   1.1756.1 Available under license

1.1757 reflectasm 1.10.1
   1.1757.1 Available under license

1.1758 babel-plugin-transform-es2015-destructuring 6.23.0
   1.1758.1 Available under license

1.1759 he 1.2.0
   1.1759.1 Available under license

1.1760 saslprep 1.0.3 1.0.3
   1.1760.1 Available under license

1.1761 atk 2.18.1-2ubuntu1
   1.1761.1 Available under license

1.1762 mesa 18.0.5-0ubuntu0~16.04.1
   1.1762.1 Available under license

1.1763 unicode-property-aliases-ecmascript 2.0.0
   1.1763.1 Available under license

1.1764 scala 2.11.12
   1.1764.1 Available under license

1.1765 nopt 3.0.6 3.0.6
   1.1765.1 Available under license

1.1766 ms 2.1.1 2.1.1
   1.1766.1 Available under license

1.1767 bson 4.2.3
   1.1767.1 Available under license

1.1768 @babel/plugin-proposal-private-methods 7.16.0
   1.1768.1 Available under license

1.1769 jetty-io 9.3.27.v20190418
   1.1769.1 Available under license

1.1770 @babel/helper-split-export-declaration 7.16.0
   1.1770.1 Available under license

1.1771 jackson-jaxrs-xml-provider 2.10.4
   1.1771.1 Available under license

1.1772 commons-dbcp 1.4
   1.1772.1 Available under license

1.1773 node-readfiles 0.2.0 0.2.0
   1.1773.1 Available under license

1.1774 libxkbcommon 0.5.0-1ubuntu2.1
1.1774.1 Available under license

1.1775 jedis 2.10.2-cisco
   1.1775.1 Available under license

1.1776 @babel/helper-remap-async-to-generator 7.16.4
   1.1776.1 Available under license

1.1777 jsr305 3.0.1

1.1778 buffer-equal-constant-time 1.0.1 1.0.1
   1.1778.1 Available under license

1.1779 babel-plugin-transform-reserved-words 7.16.0
   1.1779.1 Available under license

1.1780 node-ensure 0.0.0 0.0.0
   1.1780.1 Available under license

1.1781 yargs 13.2.4 13.2.4
   1.1781.1 Available under license

1.1782 babel-plugin-transform-es2015-template-literals 6.22.0
   1.1782.1 Available under license

1.1783 xml-apis 1.4.01
   1.1783.1 Available under license

1.1784 jetty 9.3.27.v20190418
   1.1784.1 Available under license

1.1785 babel-helper-bindify-decorators 6.24.1
   1.1785.1 Available under license

1.1786 commons-text 1.8
   1.1786.1 Available under license

1.1787 paranamer 2.3
   1.1787.1 Available under license

1.1788 eslint 3.19.0
   1.1788.1 Available under license

1.1789 jackson-databind 2.12.3
   1.1789.1 Available under license

1.1790 commons-configuration2 2.1.1
   1.1790.1 Available under license

1.1791 jaxb-api 2.2.11
   1.1791.1 Available under license

1.1792 commons-io 2.6
   1.1792.1 Available under license

1.1793 de.huxhorn.lilith.logback.converter-classic 0.9.44
   1.1793.1 Available under license

1.1794 uglify-js 3.11.3 3.11.3
1.1794.1 Available under license
1.1795 qs 6.5.2 6.5.2
1.1795.1 Available under license
1.1796 jnr-posix 1.1.4
1.1796.1 Available under license
1.1797 woodstox-core 5.0.3
1.1797.1 Available under license
1.1798 is-path-inside 3.0.3
1.1798.1 Available under license
1.1799 kotlin 1.4.0
1.1799.1 Available under license
1.1800 define-property 0.2.5 0.2.5
1.1800.1 Available under license
1.1801 negotiator 0.5.3 0.5.3
1.1801.1 Available under license
1.1802 commons-el 5.5.23
1.1802.1 Available under license
1.1803 assign-symbols 1.0.0 1.0.0
1.1803.1 Available under license
1.1804 regex-cache 0.4.4
1.1804.1 Available under license
1.1805 jna 5.8.0
1.1805.1 Available under license
1.1806 to-regex-range 2.1.1 2.1.1
1.1806.1 Available under license
1.1807 process-on-spawn 1.0.0
1.1807.1 Available under license
1.1808 generate-object-property 1.2.0 1.2.0
1.1808.1 Available under license
1.1809 type-detect 4.0.8
1.1809.1 Available under license
1.1810 component-emitter 1.1.2 1.1.2
1.1810.1 Available under license
1.1811 crypto-js 4.0.0 4.0.0
1.1811.1 Available under license
1.1812 de.huxhorn.lilith.data.converter 0.9.44
1.1812.1 Available under license
1.1813 slash 1.0.0 1.0.0
1.1813.1 Available under license
1.1814 nice-try 1.0.5 1.0.5
  1.1814.1 Available under license
1.1815 lodash.get 4.4.2 4.4.2
  1.1815.1 Available under license
1.1816 serve-static 1.14.1 1.14.1
  1.1816.1 Available under license
1.1817 drill-storage-hbase 1.19.0
  1.1817.1 Available under license
1.1818 color-convert 0.5.3 0.5.3
  1.1818.1 Available under license
1.1819 jackson-dataformat-cbor 2.6.7
  1.1819.1 Available under license
1.1820 grpc-netty-shaded 1.27.1
  1.1820.1 Available under license
1.1821 parse-color 1.0.0 1.0.0
  1.1821.1 Available under license
1.1822 escodegen 1.14.3 1.14.3
  1.1822.1 Available under license
1.1823 estraverse 4.2.0
  1.1823.1 Available under license
1.1824 is-data-descriptor 0.1.4 0.1.4
  1.1824.1 Available under license
1.1825 pre-push 0.1.1 0.1.1
  1.1825.1 Available under license
1.1826 mchange-commons-java 0.2.15
  1.1826.1 Available under license
1.1827 javaewah 0.3.2
  1.1827.1 Available under license
1.1828 jsreport-child-templates 1.4.0 1.4.0
  1.1828.1 Available under license
1.1829 fs-readdir-recursive 1.1.0
  1.1829.1 Available under license
1.1830 babel-helper-remap-async-to-generator 6.24.1
  1.1830.1 Available under license
1.1831 external-editor 3.1.0 3.1.0
  1.1831.1 Available under license
1.1832 y18n 5.0.8 5.0.8
  1.1832.1 Available under license
1.1833 x-xss-protection 1.0.0 1.0.0
1.1833.1 Available under license

1.1834 aspect-j 1.8.13
1.1834.1 Available under license

1.1835 figures 2.0.0 2.0.0
1.1835.1 Available under license

1.1836 convert-source-map 1.5.1 1.5.1
1.1836.1 Available under license

1.1837 metrics 2.2.0
1.1837.1 Available under license

1.1838 hk2-api 2.6.1
1.1838.1 Available under license

1.1839 parquet-hadoop-bundle 1.10.0
1.1839.1 Available under license

1.1840 @babel/plugin-transform-regenerator 7.16.0
1.1840.1 Available under license

1.1841 which-boxed-primitive 1.0.2
1.1841.1 Available under license

1.1842 ws 1.1.5 1.1.5
1.1842.1 Available under license

1.1843 babel-plugin-syntax-class-properties 6.13.0
1.1843.1 Available under license

1.1844 domelementtype 1.3.0 1.3.0
1.1844.1 Available under license

1.1845 immutability-helper 2.7.1 2.7.1
1.1845.1 Available under license

1.1846 htrace-core4 4.2.0-incubating
1.1846.1 Available under license

1.1847 semver 6.3.0 6.3.0
1.1847.1 Available under license

1.1848 aws-java-sdk-dynamodb 1.11.375
1.1848.1 Available under license

1.1849 jansi-osx 1.8
1.1849.1 Available under license

1.1850 archiver-utils 1.3.0 1.3.0
1.1850.1 Available under license

1.1851 process-nextick-args 2.0.1 2.0.1
1.1851.1 Available under license

1.1852 apib2swagger 1.12.0 1.12.0
1.1852.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>is-core-module</td>
<td>2.0.0 2.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>libxml</td>
<td>2.9.3+dfsg1</td>
<td>1ubuntu0.7</td>
</tr>
<tr>
<td>ajv</td>
<td>6.10.0 6.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>awaitility</td>
<td>3.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/plugin-transform-spread</td>
<td>7.16.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>batik-codec</td>
<td>1.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>parquet-column</td>
<td>1.12.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>influx</td>
<td>5.6.3 5.6.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>batik-squiggle</td>
<td>1.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>lodash-foreach</td>
<td>4.5.0 4.5.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>globals</td>
<td>9.18.0 9.18.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>istack-commons</td>
<td>3.0.7</td>
<td>Available under license</td>
</tr>
<tr>
<td>ecc-jsbn</td>
<td>0.1.2 0.1.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>maxmind-db</td>
<td>1.4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>hbase-metrics-api</td>
<td>2.4.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>jmespath-java</td>
<td>1.11.837</td>
<td>Available under license</td>
</tr>
<tr>
<td>domutils</td>
<td>1.5.1 1.5.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-runtime</td>
<td>6.26.0 6.26.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>okio</td>
<td>1.13.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-plugin-transform-es2015-parameters</td>
<td>6.24.1</td>
<td></td>
</tr>
</tbody>
</table>
1.1872.1 Available under license

1.1873 tpch-sample-data 1.19.0
1.1873.1 Available under license

1.1874.1 Available under license

1.1875 ms 0.7.2 0.7.2
1.1875.1 Available under license

1.1876 color-convert 1.9.2
1.1876.1 Available under license

1.1877 emoji-regex 7.0.3 7.0.3
1.1877.1 Available under license

1.1878 globals 11.7.0
1.1878.1 Available under license

1.1879 object.values 1.1.5
1.1879.1 Available under license

1.1880 source-map-support 0.4.18 0.4.18
1.1880.1 Available under license

1.1881 generate-function 2.3.1
1.1881.1 Available under license

1.1882 asap 2.0.6 2.0.6
1.1882.1 Available under license

1.1883 drill-format-excel 1.19.0
1.1883.1 Available under license

1.1884 prepend-http 2.0.0
1.1884.1 Available under license

1.1885 commons-compiler 3.0.11

1.1886 p-finally 1.0.0 1.0.0
1.1886.1 Available under license

1.1887 copy-descriptor 0.1.1 0.1.1
1.1887.1 Available under license

1.1888 signal-exit 3.0.3 3.0.3
1.1888.1 Available under license

1.1889 parse-filepath 1.0.2 1.0.2
1.1889.1 Available under license

1.1890 os-tmpdir 1.0.2 1.0.2
1.1890.1 Available under license

1.1891 split-on-first 1.1.0 1.1.0
1.1891.1 Available under license

1.1892 regjsgen 0.2.0
1.1892.1 Available under license

1.1893 netty-tcnative-boringssl-static 2.0.26.Final
   1.1893.1 Available under license

1.1894 cookiejar 2.1.2 2.1.2
   1.1894.1 Available under license

1.1895 event-stream 3.3.4 3.3.4
   1.1895.1 Available under license

1.1896 jnr-x86asm 1.0.2

1.1897 kafka-clients 2.8.0
   1.1897.1 Available under license

1.1898 drill-hive-exec-shaded 1.19.0
   1.1898.1 Available under license

1.1899 babel-plugin-transform-es2015-shorthand-properties 6.24.1
   1.1899.1 Available under license

1.1900 node-signpdf 1.3.2 1.3.2
   1.1900.1 Available under license

1.1901 flat 4.1.1
   1.1901.1 Available under license

1.1902 commons-pool 1.6
   1.1902.1 Available under license

1.1903 passport-http 0.3.0 0.3.0
   1.1903.1 Available under license

1.1904 detect-indent 4.0.0 4.0.0
   1.1904.1 Available under license

1.1905 metrics 4.0.2

1.1906 natty 0.13
   1.1906.1 Available under license

1.1907 ansi-styles 0.1.3 0.1.3
   1.1907.1 Available under license

1.1908 invert-kv 1.0.0 1.0.0
   1.1908.1 Available under license

1.1909 hadoop-mapreduce-client-core 3.2.2
   1.1909.1 Available under license

1.1910 crc 3.8.0 3.8.0
   1.1910.1 Available under license

1.1911 object-assign 3.0.0 3.0.0
   1.1911.1 Available under license

1.1912 locate-path 5.0.0 5.0.0
   1.1912.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>jsreport-scheduling</td>
<td>2.5.0</td>
<td></td>
</tr>
<tr>
<td>glib</td>
<td>2.48.2-0ubuntu1</td>
<td></td>
</tr>
<tr>
<td>jsreport-xlsx</td>
<td>2.5.0</td>
<td></td>
</tr>
<tr>
<td>chalk</td>
<td>1.1.3</td>
<td></td>
</tr>
<tr>
<td>glib</td>
<td>2.48.2-0ubuntu1</td>
<td></td>
</tr>
<tr>
<td>jsreport-xlsx</td>
<td>2.5.0</td>
<td></td>
</tr>
<tr>
<td>node-modules-regexp</td>
<td>1.0.0</td>
<td></td>
</tr>
<tr>
<td>chalk</td>
<td>1.1.3</td>
<td></td>
</tr>
<tr>
<td>express</td>
<td>4.16.3</td>
<td></td>
</tr>
<tr>
<td>babel-eslint</td>
<td>6.1.2</td>
<td></td>
</tr>
<tr>
<td>gtk</td>
<td>3.18.9-1ubuntu3.3</td>
<td></td>
</tr>
<tr>
<td>entities</td>
<td>1.1.2</td>
<td></td>
</tr>
<tr>
<td>is-my-json-valid</td>
<td>2.17.2</td>
<td></td>
</tr>
<tr>
<td>dom-serializer</td>
<td>0.1.1</td>
<td></td>
</tr>
<tr>
<td>drill-jdbc-all</td>
<td>1.19.0</td>
<td></td>
</tr>
<tr>
<td>react-lifecycles-compat</td>
<td>3.0.4</td>
<td></td>
</tr>
<tr>
<td>converter-jackson</td>
<td>2.8.1</td>
<td></td>
</tr>
<tr>
<td>jackson-xc</td>
<td>2.12.1</td>
<td></td>
</tr>
<tr>
<td>multer</td>
<td>1.3.1</td>
<td></td>
</tr>
</tbody>
</table>
1.1932.1 Available under license

1.1933 icu 55.1 7ubuntu0.4
1.1933.1 Available under license

1.1934 es-abstract 1.19.1
1.1934.1 Available under license

1.1935 @babel/helper-builder-binary-assignment-operator-visitor 7.16.0
1.1935.1 Available under license

1.1936 has-flag 4.0.0
1.1936.1 Available under license

1.1937 jsreport-pdf-utils 1.10.1 1.10.1
1.1937.1 Available under license

1.1938 set-value 2.0.1 2.0.1
1.1938.1 Available under license

1.1939 gson-fire 1.5.1
1.1939.1 Available under license

1.1940 metrics 4.1.0
1.1940.1 Available under license

1.1941 asm-analysis 7.3.1
1.1941.1 Available under license

1.1942 duplexer 0.1.1 0.1.1
1.1942.1 Available under license

1.1943 thrift 0.7.0
1.1943.1 Available under license

1.1944 bcrypt-pbkdf 1.0.2 1.0.2
1.1944.1 Available under license

1.1945 hawtjni-runtime 1.17
1.1945.1 Available under license

1.1946 libevent 1.4.13
1.1946.1 Available under license

1.1947 xmlbuilder 10.1.1 10.1.1
1.1947.1 Available under license

1.1948 levn 0.3.0 0.3.0
1.1948.1 Available under license

1.1949 kryo-shaded 3.0.3
1.1950 @babel/plugin-syntax-dynamic-import 7.8.3
1.1950.1 Available under license

1.1951 oauth-sign 0.9.0 0.9.0
1.1951.1 Available under license

1.1952 globals 11.12.0
1.1952.1 Available under license

1.1953 batik-xml 1.13
1.1953.1 Available under license

1.1954 mockito-core 2.23.4
1.1954.1 Available under license

1.1955 babel-plugin-transform-es2015-for-of 6.23.0 6.23.0
1.1955.1 Available under license

1.1956 jackson 2.12.1
1.1956.1 Available under license

1.1957 mongodb-driver-core 4.2.3
1.1957.1 Available under license

1.1958 babel-plugin-synt{}x20;exponentiation-operator 6.13.0 6.13.0
1.1958.1 Available under license

1.1959 filename-regex 2.0.1
1.1959.1 Available under license

1.1960 react-intl 2.4.0 2.4.0
1.1960.1 Available under license

1.1961 is-extglob 2.1.1
1.1961.1 Available under license

1.1962 ms 2.1.1
1.1962.1 Available under license

1.1963 mongodb-core 2.1.20 2.1.20
1.1963.1 Available under license

1.1964 extsprintf 1.3.0 1.3.0
1.1964.1 Available under license

1.1965 @babel/helper-module-imports 7.16.0
1.1965.1 Available under license

1.1966 jsreport-jsrender 2.1.1 2.1.1
1.1966.1 Available under license

1.1967 swagger 1.5.0
1.1967.1 Available under license

1.1968 d-bus 1.10.6-1ubuntu3.5
1.1968.1 Available under license

1.1969 jetty-servlets 9.3.27.v20190418
1.1970 netty-transport 4.0.48.Final
1.1970.1 Available under license

1.1971 kerb-identity 1.0.0
1.1971.1 Available under license

1.1972 parquet-common 1.12.0
1.1972.1 Available under license
1.1973 ncp 2.0.0 2.0.0
  1.1973.1 Available under license
1.1974 asm-all-repackaged 2.2.0-b21
  1.1974.1 Available under license
  1.1975.1 Available under license
1.1976 jetty-servlet 9.4.41.v20210516
  1.1976.1 Available under license
1.1977 is-callable 1.2.3
  1.1977.1 Available under license
1.1978 har-validator 5.1.5 5.1.5
  1.1978.1 Available under license
1.1979 aggdesigner-algorithm 6.0
1.1980 topo 1.1.0 1.1.0
  1.1980.1 Available under license
1.1981 path-is-absolute 1.0.1 1.0.1
  1.1981.1 Available under license
1.1982 jersey-mvc 2.25.1
  1.1982.1 Available under license
1.1983 babel-plugin-polyfill-regenerator 0.3.0
  1.1983.1 Available under license
1.1984 socket.io-adapter 0.5.0 0.5.0
  1.1984.1 Available under license
1.1985 jetty-util 9.4.41.v20210516
  1.1985.1 Available under license
1.1986 merge-class-names 1.3.0 1.3.0
  1.1986.1 Available under license
1.1987 asynckit 0.4.0 0.4.0
  1.1987.1 Available under license
1.1988 free-type 2.6.1 0.1ubuntu2.4
  1.1988.1 Available under license
1.1989 syslog-java-client 1.1.7
1.1990 async 2.6.3 2.6.3
  1.1990.1 Available under license
1.1991 mkdirp 0.3.0 0.3.0
  1.1991.1 Available under license
1.1992 http-errors 1.6.2 1.6.2
  1.1992.1 Available under license
1.1993 poi 5.0.0
  1.1993.1 Available under license
1.1994 path-to-regexp 0.1.7 0.1.7
  1.1994.1 Available under license
1.1995 ajv 4.11.8
  1.1995.1 Available under license
1.1996 amqp-client 4.8.0
  1.1996.1 Available under license
1.1997 arr-union 3.1.0 3.1.0
  1.1997.1 Available under license
1.1998 highlight-words-core 1.2.2 1.2.2
  1.1998.1 Available under license
1.1999 yallist 2.1.2 2.1.2
  1.1999.1 Available under license
1.2000 java-sizeof 0.0.5
  1.2000.1 Available under license
1.2001 babel-register 6.26.0
  1.2001.1 Available under license
1.2002 fsevents 2.1.3 2.1.3
  1.2002.1 Available under license
1.2003 json-refs 3.0.15 3.0.15
  1.2003.1 Available under license
1.2004 strip-ansi 6.0.1
  1.2004.1 Available under license
1.2005 deep-extend 0.6.0
  1.2005.1 Available under license
1.2006 node-releases 2.0.1
  1.2006.1 Available under license
1.2007 @babel/core 7.16.0
  1.2007.1 Available under license
1.2008 traverse 0.3.9 0.3.9
  1.2008.1 Available under license
1.2009 buffer 5.6.0 5.6.0
  1.2009.1 Available under license
1.2010 express 4.13.4 4.13.4
  1.2010.1 Available under license
1.2011 is-callable 1.2.4
  1.2011.1 Available under license
1.2012 jboss-logging 3.1.1.GA
1.2012.1 Available under license

1.2013 typedarray 0.0.6 0.0.6
1.2013.1 Available under license

1.2014 netty-transport-native-epoll 4.1.45.Final
1.2014.1 Available under license

1.2015 batik-gvt 1.13
1.2015.1 Available under license

1.2016 ion-java 1.0.2
1.2016.1 Available under license

1.2017 jctools-core 2.0.1
1.2017.1 Available under license

1.2018 expat 2.1.0 7ubuntu0.16.04.5
1.2018.1 Available under license

1.2019 assertion-error 1.1.0
1.2019.1 Available under license

1.2020 guice-servlet 4.0
1.2020.1 Available under license

1.2021 semaphore-async-await 1.5.1 1.5.1
1.2021.1 Available under license

1.2022 optionator 0.8.3
1.2022.1 Available under license

1.2023 es6-set 0.1.5
1.2023.1 Available under license

1.2024 guava 30.1.1-jre
1.2024.1 Available under license

1.2025 lodash-max 4.0.1 4.0.1
1.2025.1 Available under license

1.2026 aws-java-sdk-s3 1.11.375
1.2026.1 Available under license

1.2027 liftoff 3.0.0 3.0.0
1.2027.1 Available under license

1.2028 aircompressor 0.10
1.2028.1 Available under license

1.2029 rimraf 2.6.3
1.2029.1 Available under license

1.2030 commons-validator 1.7
1.2030.1 Available under license

1.2031 mime-types 2.1.19 2.1.19
1.2031.1 Available under license
1.2032 is-npm 5.0.0
   1.2032.1 Available under license
1.2033 connect 3.4.1 3.4.1
   1.2033.1 Available under license
1.2034 calcite-core 1.21.0-drill-r2
   1.2034.1 Available under license
1.2035 yauzl 2.10.0 2.10.0
   1.2035.1 Available under license
1.2036 ant-launcher 1.10.9
   1.2036.1 Available under license
1.2037 log-symbols 3.0.0
   1.2037.1 Available under license
1.2038 swagger-core 2.0.7
   1.2038.1 Available under license
1.2039 js-cookie 2.1.1 2.1.1
   1.2039.1 Available under license
1.2040 colord 1.2.12-1ubuntu1
   1.2040.1 Available under license
1.2041 is-unc-path 1.0.0 1.0.0
   1.2041.1 Available under license
1.2042 object-copy 0.1.0 0.1.0
   1.2042.1 Available under license
1.2043 method-override 2.3.10 2.3.10
   1.2043.1 Available under license
1.2044 xmpcore 6.0.6
   1.2044.1 Available under license
1.2045 locate-path 3.0.0
   1.2045.1 Available under license
1.2046 jersey-server 2.25.1
   1.2046.1 Available under license
1.2047 is-my-ip-valid 1.0.0
   1.2047.1 Available under license
1.2048 camelcase 6.2.1
   1.2048.1 Available under license
1.2049 wrap-ansi 2.1.0 2.1.0
   1.2049.1 Available under license
1.2050 istanbul-lib-source-maps 4.0.0
   1.2050.1 Available under license
1.2051 guice-assistedinject 3.0
1.2051.1 Available under license

1.2052 oas-kit-common 1.0.8 1.0.8
1.2052.1 Available under license

1.2053 liquibase 3.8.7
1.2053.1 Available under license

1.2054 @babel/code-frame 7.10.4
1.2054.1 Available under license

1.2055 @babel/generator 7.11.5
1.2055.1 Available under license

1.2056 kotlin 1.4.10
1.2056.1 Available under license

1.2057 kerby-xdr 1.0.0
1.2057.1 Available under license

1.2058 caller-path 0.1.0
1.2058.1 Available under license

1.2059 babel-helper-define-map 6.26.0 6.26.0
1.2059.1 Available under license

1.2060 eslint-config-airbnb-base 9.0.0
1.2060.1 Available under license

1.2061 fill-range 7.0.1 7.0.1
1.2061.1 Available under license

1.2062 mime 1.4.1 1.4.1
1.2062.1 Available under license

1.2063 minimist 1.2.0 1.2.0
1.2063.1 Available under license

1.2064 is-windows 1.0.2 1.0.2
1.2064.1 Available under license

1.2065 @babel/helpers 7.16.3
1.2065.1 Available under license

1.2066 @babel/template 7.16.0
1.2066.1 Available under license

1.2067 pause-stream 0.0.11 0.0.11
1.2067.1 Available under license

1.2068 for-own 0.1.5
1.2068.1 Available under license

1.2069 bouncy-castle 1.60
1.2069.1 Available under license

1.2070 lodash.curry 4.1.1 4.1.1
1.2070.1 Available under license
1.2071 xmlgraphics-commons 2.4
  1.2071.1 Available under license
1.2072 jetty-continuation 9.4.41.v20210516
  1.2072.1 Available under license
1.2073 har-validator 5.0.3 5.0.3
  1.2073.1 Available under license
1.2074 json-schema-compatibility 1.1.0 1.1.0
  1.2074.1 Available under license
1.2075 is-glob 4.0.3
  1.2075.1 Available under license
1.2076 ansi-colors 3.2.3
  1.2076.1 Available under license
1.2077 jul-to-slf4j 1.7.5
1.2078 d 1.0.1
  1.2078.1 Available under license
1.2079 quartz 2.3.2
  1.2079.1 Available under license
1.2080 jersey-hk2 2.32
  1.2080.1 Available under license
1.2081 batik-swing 1.13
  1.2081.1 Available under license
1.2082 xmlenc 0.52
  1.2082.1 Available under license
1.2083 javax-inject 2.5.0-b32
  1.2083.1 Available under license
1.2084 drifter.js 2.6.7 2.6.7
  1.2084.1 Available under license
1.2085 is-accessor-descriptor 0.1.6 0.1.6
  1.2085.1 Available under license
1.2086 backport-util-concurrent 3.1
  1.2086.1 Available under license
1.2087 commons-http-client 3.1
  1.2087.1 Available under license
1.2088 avro-ipc 1.9.1
  1.2088.1 Available under license
1.2089 cookie 0.3.1 0.3.1
  1.2089.1 Available under license
1.2090 bl 2.2.1 2.2.1
  1.2090.1 Available under license
1.2091 which 2.0.2
  1.2091.1 Available under license
1.2092 secure-keys 1.0.0 1.0.0
  1.2092.1 Available under license
1.2093 datanucleus-rdbms 4.1.19
  1.2093.1 Available under license
1.2094 restore-cursor 1.0.1
  1.2094.1 Available under license
1.2095 uuid 3.4.0 3.4.0
  1.2095.1 Available under license
1.2096 regenerator-transform 0.10.1
  1.2096.1 Available under license
1.2097 babel-core 6.26.3
  1.2097.1 Available under license
1.2098 sax 1.2.4 1.2.4
  1.2098.1 Available under license
1.2099 ip-regex 1.0.3
  1.2099.1 Available under license
1.2100 jsreport-office 1.2.0 1.2.0
  1.2100.1 Available under license
1.2101 de.huxhorn.lilith.data.logging.protobuf 0.9.44
  1.2101.1 Available under license
1.2102 vary 1.1.2 1.1.2
  1.2102.1 Available under license
1.2103 private 0.1.8 0.1.8
  1.2103.1 Available under license
1.2104 aws4 1.8.0 1.8.0
  1.2104.1 Available under license
1.2105 mime-db 1.35.0 1.35.0
  1.2105.1 Available under license
1.2106 pcre 2.48.2-0ubuntu4.4
  1.2106.1 Available under license
1.2107 hbase-mapreduce 2.0.0-alpha4
  1.2107.1 Available under license
1.2108 regjsgen 0.2.0 0.2.0
  1.2108.1 Available under license
1.2109 strip-json-comments 2.0.1
  1.2109.1 Available under license
1.2110 yargs 15.4.1 15.4.1
1.210.1 Available under license
1.211 asm 3.3.1
1.211.1 Available under license
1.212 optionator 0.8.3 0.8.3
1.212.1 Available under license
1.213 ansi-regex 5.0.1
1.213.1 Available under license
1.214 kind-of 3.2.2
1.214.1 Available under license
1.215 commons-beanutils 1.8.0
1.215.1 Available under license
1.216 tsconfig-paths 3.12.0
1.216.1 Available under license
1.217 calcite-elasticsearch 1.21.0-drill-r2
1.217.1 Available under license
1.218 cli-boxes 2.2.1
1.218.1 Available under license
1.219 urijs 1.19.6 1.19.6
1.219.1 Available under license
1.220 babel-plugin-syntax-nullish-coalescing-operator 7.8.3
1.220.1 Available under license
1.221 p-locate 3.0.0
1.221.1 Available under license
1.222 combined-stream 1.0.6 1.0.6
1.222.1 Available under license
1.223 websocket-common 9.3.27.v20190418
1.224 swagger2openapi 7.0.3 7.0.3
1.224.1 Available under license
1.225 shelljs 0.7.8
1.225.1 Available under license
1.226 @babel/helper-hoist-variables 7.16.0
1.226.1 Available under license
1.227 lodash.reduce 4.5.0 4.5.0
1.227.1 Available under license
1.228 detect-indent 4.0.0
1.228.1 Available under license
1.229classnames 2.2.6 2.2.6
1.229.1 Available under license
1.230 okhttp 4.9.1
1.2130.1 Available under license

1.2131 abbrev 1.0.9
1.2131.1 Available under license

1.2132 jackson 2.4.0
1.2132.1 Available under license

1.2133 ajv-keywords 3.2.0 3.2.0
1.2133.1 Available under license

1.2134 zstd-jni 1.4.9-1
1.2134.1 Available under license

1.2135 format-util 1.0.5 1.0.5
1.2135.1 Available under license

1.2136 ws 6.1.4 6.1.4
1.2136.1 Available under license

1.2137 regenerator-runtime 0.10.5
1.2137.1 Available under license

1.2138 jffi 1.2.15
1.2138.1 Available under license

1.2139 babylon 6.18.0 6.18.0
1.2139.1 Available under license

1.2140 libxi 1.7.6-1
1.2140.1 Available under license

1.2141 install 0.12.2 0.12.2
1.2141.1 Available under license

1.2142 commons-io 2.7
1.2142.1 Available under license

1.2143 esprima 4.0.1
1.2143.1 Available under license

1.2144 mylib 0.0.0
1.2144.1 Available under license

1.2145.1 Available under license

1.2146 yeast 0.1.2 0.1.2
1.2146.1 Available under license

1.2147 de.huxhorn.lilith.data.eventsource 0.9.44
1.2147.1 Available under license

1.2148 strip-ansi 6.0.0
1.2148.1 Available under license

1.2149 zookeeper-jute 3.5.7
1.2149.1 Available under license
1.2150 pinkie 1.0.0 1.0.0
   1.2150.1 Available under license
1.2151 @babel/plugin-proposal-class-properties 7.16.0
   1.2151.1 Available under license
1.2152 is-buffer 2.0.4 2.0.4
   1.2152.1 Available under license
1.2153 wrap-ansi 6.2.0
   1.2153.1 Available under license
1.2154 dont-sniff-mimetype 1.0.0 1.0.0
   1.2154.1 Available under license
1.2155 pdfjs-dist 2.0.305 2.0.305
   1.2155.1 Available under license
1.2156 libdrm 2.4.91 2~16.04.1
1.2157 follow-redirects 1.5.10 1.5.10
   1.2157.1 Available under license
1.2158 drill-format-hdf5 1.19.0
   1.2158.1 Available under license
1.2159 libvpx 1.7.0-1367-gac3eccdc2
   1.2159.1 Available under license
1.2160 calcite-example-function 1.21.0
   1.2160.1 Available under license
1.2161 decompress-response 3.3.0
   1.2161.1 Available under license
1.2162 hive-metastore 3.1.2
   1.2162.1 Available under license
1.2163 wrap-ansi 7.0.0 7.0.0
   1.2163.1 Available under license
1.2164 function-bind 1.1.1 1.1.1
   1.2164.1 Available under license
1.2165 findup-sync 3.0.0 3.0.0
   1.2165.1 Available under license
1.2166 hive-shims-common 3.1.2
   1.2166.1 Available under license
1.2167 yargs-parser 18.1.3 18.1.3
   1.2167.1 Available under license
1.2168 hive-standalone-metastore 3.1.2
   1.2168.1 Available under license
1.2169 request-promise-core 1.1.4 1.1.4
   1.2169.1 Available under license
1.2170 javax.servlet.jsp 2.3.2
   1.2170.1 Available under license

1.2171 javax-inject 1
   1.2171.1 Available under license

1.2172 hive-exec 3.1.2
   1.2172.1 Available under license

1.2173 yargs-parser 13.1.2
   1.2173.1 Available under license

1.2174 setprototypeof 1.1.0 1.1.0
   1.2174.1 Available under license

1.2175 debug 3.2.7
   1.2175.1 Available under license

1.2176 ansi-align 3.0.0
   1.2176.1 Available under license

1.2177 commons-pool 2.4.3
   1.2177.1 Available under license

1.2178 hbase-shaded-protobuf 3.4.1
   1.2178.1 Available under license

1.2179 require-main-filename 2.0.0 2.0.0
   1.2179.1 Available under license

1.2180 stream-combiner 0.0.4 0.0.4
   1.2180.1 Available under license

1.2181 log4j-slf4j-impl 2.17.1
   1.2181.1 Available under license

1.2182 es6-promisify 5.0.0 5.0.0
   1.2182.1 Available under license

1.2183 semver 7.0.0
   1.2183.1 Available under license

1.2184 is-regex 1.1.4
   1.2184.1 Available under license

1.2185 cors 2.8.5 2.8.5
   1.2185.1 Available under license

1.2186 jetty-io 9.4.34.v20201102
   1.2186.1 Available under license

1.2187 jsrender 1.0.4 1.0.4
   1.2187.1 Available under license

1.2188 type-of-is 3.5.1 3.5.1
   1.2188.1 Available under license

1.2189 which 1.3.1
1.2189.1 Available under license
1.2190 kind-of 6.0.3 6.0.3
   1.2190.1 Available under license
1.2191 p-map 3.0.0
   1.2191.1 Available under license
1.2192 hdrhistogram 2.1.12
   1.2192.1 Available under license
1.2193 open-csv 5.4
   1.2193.1 Available under license
1.2194 performance-now 2.1.0 2.1.0
   1.2194.1 Available under license
1.2195 raml-to-swagger 1.1.0 1.1.0
   1.2195.1 Available under license
1.2196 mimic-fn 2.1.0 2.1.0
   1.2196.1 Available under license
1.2197 parseuri 0.0.6 0.0.6
   1.2197.1 Available under license
1.2198 babel-plugin-transform-runtime 6.23.0
   1.2198.1 Available under license
1.2199 jcommander 1.78
   1.2199.1 Available under license
1.2200 uuid 3.2.1 3.2.1
   1.2200.1 Available under license
1.2201 @babel/plugin-transform-modules-systemjs 7.16.0
   1.2201.1 Available under license
1.2202 type 1.2.0
   1.2202.1 Available under license
1.2203 drill-mongo-storage 1.19.0
   1.2203.1 Available under license
1.2204 babel-helper-optimise-call-expression 6.24.1 6.24.1
   1.2204.1 Available under license
1.2205 for-own 1.0.0 1.0.0
   1.2205.1 Available under license
   1.2206.1 Available under license
1.2207 opentracing-noop 0.31.0
   1.2207.1 Available under license
1.2208 got 2.4.0 2.4.0
   1.2208.1 Available under license
1.2209 babel-preset-env 1.7.0
  1.2209.1 Available under license

1.2210 repeating 2.0.1
  1.2210.1 Available under license

1.2211 babel-plugin-transform-es2015-spread 6.22.0
  1.2211.1 Available under license

1.2212 jna 3.5.1
  1.2212.1 Available under license

1.2213 jsreport-resources 2.1.0 2.1.0
  1.2213.1 Available under license

1.2214 babel-template 6.26.0
  1.2214.1 Available under license

1.2215 jsbn 1.1.0 1.1.0
  1.2215.1 Available under license

1.2216 anymatch 3.1.2
  1.2216.1 Available under license

1.2217 find-up 3.0.0 3.0.0
  1.2217.1 Available under license

1.2218 netty-handler 4.1.59.Final
  1.2218.1 Available under license

1.2219 rimraf 3.0.2
  1.2219.1 Available under license

1.2220 base 0.11.2 0.11.2
  1.2220.1 Available under license

1.2221 is-glob 2.0.1
  1.2221.1 Available under license

1.2222 bitsyntax 0.1.0 0.1.0
  1.2222.1 Available under license

1.2223 ansi-regexp 5.0.0
  1.2223.1 Available under license

1.2224 uzaygezen-core 0.2
  1.2224.1 Available under license

1.2225 babel-plugin-transform-es2015-arrow-functions 6.22.0 6.22.0
  1.2225.1 Available under license

1.2226 @babel/plugin-proposal-json-strings 7.16.0
  1.2226.1 Available under license

1.2227 anymatch 3.1.1
  1.2227.1 Available under license

1.2228 commons-lang3 3.9
1.2228.1 Available under license

1.2229 **compressible 2.0.14** 2.0.14
   1.2229.1 Available under license

1.2230 **js-yaml 3.14.0**
   1.2230.1 Available under license

1.2231 **type 2.1.0**
   1.2231.1 Available under license

1.2232 **p-locate 2.0.0**
   1.2232.1 Available under license

1.2233 **minio 4.0.0**
1.2234 **fast-levenshtein 2.0.6** 2.0.6
   1.2234.1 Available under license

1.2235 **range-parser 1.2.1** 1.2.1
   1.2235.1 Available under license

1.2236 **default-require-extensions 3.0.0**
   1.2236.1 Available under license

1.2237 **tephra-api 0.6.0**
   1.2237.1 Available under license

1.2238 **cookie 0.4.0** 0.4.0
   1.2238.1 Available under license

1.2239 **doctrine 2.1.0**
   1.2239.1 Available under license

1.2240 **librest 0.7.93-1**
   1.2240.1 Available under license

1.2241 **repeat-element 1.1.3** 1.1.3
   1.2241.1 Available under license

1.2242 **jersey-container-servlet-core 2.5.1**
   1.2242.1 Available under license

1.2243 **trim-right 1.0.1** 1.0.1
   1.2243.1 Available under license

1.2244 **pango 1.38.1-1**
   1.2244.1 Available under license

1.2245 **jersey 2.25.1**
   1.2245.1 Available under license

1.2246 **http-status-codes 1.4.0** 1.4.0
   1.2246.1 Available under license

1.2247 **cglib 2.2.0-b21**
   1.2247.1 Available under license

1.2248 **snmp-4j 2.6.3**
1.2248.1 Available under license
1.2249 javax-annotation-api 1.2
   1.2249.1 Available under license
1.2250 https-proxy-agent 2.2.4
   1.2250.1 Available under license
1.2251 commons-compress 2.1.1 2.1.1
   1.2251.1 Available under license
1.2252 lcid 2.0.0 2.0.0
   1.2252.1 Available under license
1.2253 libedit 3.1-20150325-1ubuntu2
   1.2253.1 Available under license
1.2254 global-dirs 3.0.0
   1.2254.1 Available under license
1.2255 http-errors 1.6.3 1.6.3
   1.2255.1 Available under license
1.2256 orc-tools 1.5.6
   1.2256.1 Available under license
1.2257 hive-llap-common 3.1.2
   1.2257.1 Available under license
1.2258 cookiejar 2.0.6
   1.2258.1 Available under license
1.2259 array prototype flat 1.2.5
   1.2259.1 Available under license
1.2260 @babel/preset-modules 0.1.5
   1.2260.1 Available under license
1.2261 ratelimitj-inmemory 0.4.0
   1.2261.1 Available under license
1.2262 jets3t 0.6.1
   1.2262.1 Available under license
1.2263 typetools 0.5.0
   1.2263.1 Available under license
1.2264 get-caller-file 2.0.5 2.0.5
   1.2264.1 Available under license
1.2265 intl-messageformat 2.2.0 2.2.0
   1.2265.1 Available under license
1.2266 progress 1.1.8
   1.2266.1 Available under license
1.2267 jsreport-browser-client 2.2.2 2.2.2
   1.2267.1 Available under license
<table>
<thead>
<tr>
<th>1.2268</th>
<th>jsreport-docx 2.9.0 2.9.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2268.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2269</th>
<th>hbase-shaded-jersey 3.4.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2269.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2270</th>
<th>jsbn 0.1.1 0.1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2270.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2271</th>
<th>dbf-reader 0.3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2271.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2272</th>
<th>finalhandler 1.1.2 1.1.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2272.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2273</th>
<th>babel-generator 6.26.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2273.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2274</th>
<th>@babel/plugin-transform-shorthand-properties 7.16.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2274.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2275</th>
<th>node-extend 2.0.2 2.0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2275.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2276</th>
<th>babel-helper-create-class-features-plugin 7.16.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2276.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2277</th>
<th>babel-helper-explode-assignable-expression 6.24.1 6.24.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2277.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2278</th>
<th>has-symbols 1.0.1 1.0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2278.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2279</th>
<th>@babel/helper-explode-assignable-expression 7.16.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2279.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2280</th>
<th>sshpk 1.16.1 1.16.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2280.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2281</th>
<th>symbol-observable 1.2.0 1.2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2281.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2282</th>
<th>httpcomponents-client 4.5.13</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2282.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2283</th>
<th>parquet-common 1.8.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2283.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2284</th>
<th>jwa 1.1.6 1.1.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2284.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2285</th>
<th>slf4j-log4j 1.7.32</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2285.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2286</th>
<th>semver-diff 3.1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2286.1</td>
<td>Available under license</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2287</th>
<th>js-tokens 3.0.2</th>
</tr>
</thead>
</table>
1.2287.1 Available under license
1.2288 scala 2.11.0
1.2288.1 Available under license
1.2289 react-ace 5.10.0 5.10.0
1.2289.1 Available under license
1.2290 parquet-jackson 1.10.0
1.2290.1 Available under license
1.2291 istanbuljs-schema 0.1.3
1.2291.1 Available under license
1.2292 univocity-parsers 2.8.3
1.2292.1 Available under license
1.2293 client-sessions 0.8.0 0.8.0
1.2293.1 Available under license
1.2294 is-number 4.0.0
1.2294.1 Available under license
1.2295 extglob 0.3.2
1.2295.1 Available under license
1.2296 html-escaper 2.0.2
1.2296.1 Available under license
1.2297 esutils 2.0.2 2.0.2
1.2297.1 Available under license
1.2298 prelude-ls 1.1.2
1.2298.1 Available under license
1.2299 ms 2.1.3
1.2299.1 Available under license
1.2300 set-immediate-shim 1.0.1 1.0.1
1.2300.1 Available under license
1.2301 write 0.2.1
1.2301.1 Available under license
1.2302 twill-api 0.6.0-incubating
1.2302.1 Available under license
1.2303 nock 11.9.1
1.2303.1 Available under license
1.2304 txw2 2.3.1
1.2304.1 Available under license
1.2305 on-finished 2.3.0 2.3.0
1.2305.1 Available under license
1.2306 libproxy 0.4.11-ubuntu1
1.2306.1 Available under license
1.2307 jetty-http-spi 9.3.27.v20190418
1.2308 is-callable 1.1.4
   1.2308.1 Available under license
1.2309 shallow-clone 3.0.1
   1.2309.1 Available under license
1.2310 codemodel 2.6
1.2311 snake-yaml 1.27
   1.2311.1 Available under license
1.2312 calcite-cassandra 1.21.0-drill-r2
   1.2312.1 Available under license
1.2313 argparse 1.0.10
   1.2313.1 Available under license
1.2314 sprintf-js 1.0.3
   1.2314.1 Available under license
1.2315 node-promise 0.5.13 0.5.13
   1.2315.1 Available under license
1.2316 underscore 1.13.0-2 1.13.0-2
   1.2316.1 Available under license
1.2317 mute-stream 0.0.5
   1.2317.1 Available under license
1.2318 chokidar 3.3.1 3.3.1
   1.2318.1 Available under license
1.2319 cliui 4.1.0 4.1.0
   1.2319.1 Available under license
1.2320 babel-plugin-transform-es2015-computed-properties 6.24.1 6.24.1
   1.2320.1 Available under license
1.2321 jsreport-chrome-pdf 1.10.0 1.10.0
   1.2321.1 Available under license
1.2322 xalan 2.7.2
   1.2322.1 Available under license
1.2323 openssl 1.0.2m
   1.2323.1 Available under license
1.2324 sqlite 3.30.1
1.2325 nanomatch 1.2.13 1.2.13
   1.2325.1 Available under license
1.2326 fontbox 2.0.22
   1.2326.1 Available under license
1.2327 yallist 4.0.0
   1.2327.1 Available under license
<table>
<thead>
<tr>
<th>1.2328</th>
<th>axios 0.21.4 0.21.4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.2328.1 Available under license</td>
</tr>
<tr>
<td>1.2329</td>
<td>chalk 2.4.2 2.4.2</td>
</tr>
<tr>
<td></td>
<td>1.2329.1 Available under license</td>
</tr>
<tr>
<td>1.2330</td>
<td>ecdsa-sig-formatter 1.0.11 1.0.11</td>
</tr>
<tr>
<td></td>
<td>1.2330.1 Available under license</td>
</tr>
<tr>
<td>1.2331</td>
<td>mimic-response 2.1.0 2.1.0</td>
</tr>
<tr>
<td></td>
<td>1.2331.1 Available under license</td>
</tr>
<tr>
<td>1.2332</td>
<td>ansi-regex 4.1.0</td>
</tr>
<tr>
<td></td>
<td>1.2332.1 Available under license</td>
</tr>
<tr>
<td>1.2333</td>
<td>estraverse 1.9.3</td>
</tr>
<tr>
<td></td>
<td>1.2333.1 Available under license</td>
</tr>
<tr>
<td>1.2334</td>
<td>kafka-schema-registry-client 6.1.1</td>
</tr>
<tr>
<td></td>
<td>1.2334.1 Available under license</td>
</tr>
<tr>
<td>1.2335</td>
<td>jackson-jaxrs-base 2.8.4</td>
</tr>
<tr>
<td></td>
<td>1.2335.1 Available under license</td>
</tr>
<tr>
<td></td>
<td>1.2336.1 Available under license</td>
</tr>
<tr>
<td>1.2337</td>
<td>twill-discovery-api 0.6.0-incubating</td>
</tr>
<tr>
<td></td>
<td>1.2337.1 Available under license</td>
</tr>
<tr>
<td>1.2338</td>
<td>drill-storage-kafka 1.19.0</td>
</tr>
<tr>
<td></td>
<td>1.2338.1 Available under license</td>
</tr>
<tr>
<td>1.2339</td>
<td>map-visit 1.0.0 1.0.0</td>
</tr>
<tr>
<td></td>
<td>1.2339.1 Available under license</td>
</tr>
<tr>
<td>1.2340</td>
<td>kudu-client 1.14.0</td>
</tr>
<tr>
<td></td>
<td>1.2340.1 Available under license</td>
</tr>
<tr>
<td>1.2341</td>
<td>hadoop-hdfs-client 3.2.2</td>
</tr>
<tr>
<td></td>
<td>1.2341.1 Available under license</td>
</tr>
<tr>
<td>1.2342</td>
<td>jersey-common 2.25.1</td>
</tr>
<tr>
<td></td>
<td>1.2342.1 Available under license</td>
</tr>
<tr>
<td>1.2343</td>
<td>jackson-jaxrs 2.10.4</td>
</tr>
<tr>
<td></td>
<td>1.2343.1 Available under license</td>
</tr>
<tr>
<td>1.2344</td>
<td>caniuse-lite 1.0.30001202</td>
</tr>
<tr>
<td></td>
<td>1.2344.1 Available under license</td>
</tr>
<tr>
<td>1.2345</td>
<td>jline 2.9</td>
</tr>
<tr>
<td>1.2346</td>
<td>netty 4.1.61.Final</td>
</tr>
<tr>
<td></td>
<td>1.2346.1 Available under license</td>
</tr>
<tr>
<td>1.2347</td>
<td>esri-geometry-api 2.2.0</td>
</tr>
<tr>
<td></td>
<td>1.2347.1 Available under license</td>
</tr>
</tbody>
</table>
1.2348 engine.io-parser 2.2.1 2.2.1
   1.2348.1 Available under license
1.2349 disruptor 3.4.2
   1.2349.1 Available under license
1.2350 jackson-databind 2.10.5.1
   1.2350.1 Available under license
1.2351 punycode 1.4.1 1.4.1
   1.2351.1 Available under license
1.2352 util-deprecate 1.0.2 1.0.2
   1.2352.1 Available under license
1.2353 arraybuffer.slice 0.0.6 0.0.6
   1.2353.1 Available under license
1.2354 deep-eql 3.0.1
   1.2354.1 Available under license
1.2355 react-base16-styling 0.6.0 0.6.0
   1.2355.1 Available under license
1.2356 thrift 0.9.3
   1.2356.1 Available under license
1.2357 from 0.1.7 0.1.7
   1.2357.1 Available under license
1.2358 string-prototype-trimend 1.0.4
   1.2358.1 Available under license
1.2359 jetty-util 6.1.26
1.2360 annotations 4.1.1.4
   1.2360.1 Available under license
1.2361 commons-jexl 2.1.1
   1.2361.1 Available under license
1.2362 junit-platform-engine 1.7.0
   1.2362.1 Available under license
1.2363 isarray 2.0.1 2.0.1
   1.2363.1 Available under license
1.2364 array-includes 3.1.4
   1.2364.1 Available under license
1.2365 unorm 1.6.0 1.6.0
   1.2365.1 Available under license
1.2366 figures 1.7.0
   1.2366.1 Available under license
1.2367 media-typer 0.3.0 0.3.0
   1.2367.1 Available under license
1.2368 @babel/plugin-transform-sticky-regex 7.16.0
   1.2368.1 Available under license
1.2369 es6-weak-map 2.0.3
   1.2369.1 Available under license
1.2370 arr-flatten 1.1.0 1.1.0
   1.2370.1 Available under license
1.2371 yargs-parser 20.2.7 20.2.7
   1.2371.1 Available under license
1.2372 ienoopen 1.0.0 1.0.0
   1.2372.1 Available under license
1.2373 duplexer3 0.1.4
   1.2373.1 Available under license
1.2374 zlib 1.2.5
   1.2374.1 Available under license
1.2375 object-inspect 1.9.0
   1.2375.1 Available under license
1.2376 urix 0.1.0 0.1.0
   1.2376.1 Available under license
1.2377 babel-plugin-transform-class-properties 6.24.1
   1.2377.1 Available under license
1.2378 snake-yaml 1.12
   1.2378.1 Available under license
1.2379 babel-helper-call-delegate 6.24.1 6.24.1
   1.2379.1 Available under license
1.2380 @babel/plugin-transform-exponentiation-operator 7.16.0
   1.2380.1 Available under license
1.2381 jersey-guava 2.25.1
   1.2381.1 Available under license
1.2382 source-map-resolve 0.5.3 0.5.3
   1.2382.1 Available under license
1.2383 objenesis 2.6
   1.2383.1 Available under license
1.2384 @babel/plugin-syntax-class-properties 7.12.13
   1.2384.1 Available under license
1.2385 mkdirp 0.5.1 0.5.1
   1.2385.1 Available under license
1.2386 config 1.0.0
   1.2386.1 Available under license
1.2387 hoek 2.16.3 2.16.3
1.2387.1 Available under license
1.2388 resource locator 1.0.1
  1.2388.1 Available under license
1.2389 hbase-client 2.4.2
  1.2389.1 Available under license
1.2390 sway 2.0.6 2.0.6
  1.2390.1 Available under license
1.2391 http-errors 1.7.2 1.7.2
  1.2391.1 Available under license
1.2392 dayjs 1.9.3 1.9.3
  1.2392.1 Available under license
1.2393 hbase-hadoop-compat 2.4.2
  1.2393.1 Available under license
1.2394 @babel/plugin-transform-typeof-symbol 7.16.0
  1.2394.1 Available under license
1.2395 has-bigints 1.0.1
  1.2395.1 Available under license
1.2396 batik-gui-util 1.13
  1.2396.1 Available under license
1.2397 fragment-cache 0.2.1 0.2.1
  1.2397.1 Available under license
1.2398 combined-stream 1.0.8 1.0.8
  1.2398.1 Available under license
1.2399 inherits 2.0.4 2.0.4
  1.2399.1 Available under license
1.2400 istanbul-lib-processinfo 2.0.2
  1.2400.1 Available under license
1.2401 babel-plugin-transform-member-expression-literals 7.16.0
  1.2401.1 Available under license
1.2402 @babel/plugin-syntax-private-property-in-object 7.14.5
  1.2402.1 Available under license
1.2403 commons-io 1.3.2
  1.2403.1 Available under license
1.2404 color-name 1.1.3 1.1.3
  1.2404.1 Available under license
1.2405 file-entry-cache 2.0.0
  1.2405.1 Available under license
1.2406 source-map-support 0.4.18
  1.2406.1 Available under license
1.2407 jline-style 3.12.1
  1.2407.1 Available under license
1.2408 xtend 2.1.2 2.1.2
  1.2408.1 Available under license
1.2409 babel-plugin-transform-es2015-computed-properties 6.24.1
  1.2409.1 Available under license
1.2410 joda-time 2.10.5
  1.2410.1 Available under license
1.2411 hbase-zookeeper 2.4.2
  1.2411.1 Available under license
1.2412 asm-commons 7.3.1
  1.2412.1 Available under license
1.2413 libcroco 0.6.11 1
  1.2413.1 Available under license
1.2414 ejs 2.6.2 2.6.2
  1.2414.1 Available under license
1.2415 yargs 13.3.2
  1.2415.1 Available under license
1.2416 mockito 2.23.4
  1.2416.1 Available under license
1.2417 proxy-addr 2.0.4 2.0.4
  1.2417.1 Available under license
1.2418 is-accessor-descriptor 1.0.0
  1.2418.1 Available under license
1.2419 is-installed-globally 0.4.0
  1.2419.1 Available under license
1.2420 jsreport-debug 2.1.3 2.1.3
  1.2420.1 Available under license
1.2421 caching-transform 4.0.0
  1.2421.1 Available under license
1.2422 responselike 1.0.2
  1.2422.1 Available under license
1.2423 har-schema 2.0.0 2.0.0
  1.2423.1 Available under license
1.2424 jesc 1.3.0 1.3.0
  1.2424.1 Available under license
1.2425 define-properties 1.1.3 1.1.3
  1.2425.1 Available under license
1.2426 optional-require 1.0.3 1.0.3
Open Source Used In DNAC Platform Arica 125

1.2426.1 Available under license
1.2427 babel-traverse 6.26.0 6.26.0
   1.2427.1 Available under license
1.2428 ant 1.10.9
   1.2428.1 Available under license
1.2429 dom-serializer 0.1.0 0.1.0
   1.2429.1 Available under license
1.2430 stax-ex 1.8
1.2431 jasper-runtime 5.5.12
   1.2431.1 Available under license
1.2432 v8flags 2.1.1
   1.2432.1 Available under license
   1.2433.1 Available under license
1.2434 blob 0.0.4 0.0.4
   1.2434.1 Available under license
1.2435 extend-shallow 2.0.1 2.0.1
   1.2435.1 Available under license
1.2436 javax-jdo 3.2.0-m3
   1.2436.1 Available under license
1.2437 chalk 1.1.3
   1.2437.1 Available under license
1.2438 react-tagsinput 3.19.0 3.19.0
   1.2438.1 Available under license
1.2439 is-regex 1.1.2
   1.2439.1 Available under license
1.2440 is-descriptor 0.1.6
   1.2440.1 Available under license
1.2441 is-symbol 1.0.3
   1.2441.1 Available under license
1.2442 async 0.9.2 0.9.2
   1.2442.1 Available under license
1.2443 reftools 1.1.8 1.1.8
   1.2443.1 Available under license
1.2444 jmockit 1.47
   1.2444.1 Available under license
1.2445 snapdragon 0.8.2 0.8.2
   1.2445.1 Available under license
1.2446 is-absolute 1.0.0 1.0.0
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>open source used in DNAC platform</td>
<td>Arica 126</td>
<td>Available under license</td>
</tr>
<tr>
<td>object-keys</td>
<td>0.4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>jackson-datatype-joda</td>
<td>2.12.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>netty</td>
<td>4.1.56.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>jarchivelib</td>
<td>1.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>batik-transcoder</td>
<td>1.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>jsonfile</td>
<td>2.4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>anymatch</td>
<td>3.1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>parquet-column</td>
<td>1.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>jackson-annotations</td>
<td>2.4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/plugin-proposal-class-static-block</td>
<td>7.16.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-code-frame</td>
<td>6.26.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>is-equal-shallow</td>
<td>0.1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>node-fetch</td>
<td>1.7.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>tmp</td>
<td>0.0.33</td>
<td>Available under license</td>
</tr>
<tr>
<td>natural-compare</td>
<td>1.4.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>readdirp</td>
<td>2.2.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>drill-memory-base</td>
<td>1.19.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>jackson-dataformat-yaml</td>
<td>2.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>resolve-from</td>
<td>5.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>Package Name</td>
<td>Version</td>
<td>License</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>---------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>commons-configuration</td>
<td>2.1.1</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>ext</td>
<td>1.4.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>flux</td>
<td>3.1.3</td>
<td>3.1.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>normalize-path</td>
<td>3.0.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>parquet-hadoop</td>
<td>1.12.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>node-fetch-h</td>
<td>2.3.0</td>
<td>2.3.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>is-boolean-object</td>
<td>1.1.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>sparsebitset</td>
<td>1.2</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>lodashuniq</td>
<td>4.5.0</td>
<td>4.5.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>hbase-protocol-shaded</td>
<td>2.4.2</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>babel-plugin-transform-es2015-name</td>
<td>6.24.1</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>pluralize</td>
<td>1.1.6</td>
<td>1.1.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>is-glob</td>
<td>4.0.1</td>
<td>4.0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>remove-trailing-separator</td>
<td>1.1.0</td>
<td>1.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>kind-of</td>
<td>5.1.0</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>hoist-non-react-statics</td>
<td>2.5.5</td>
<td>2.5.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>de.huxhorn.lilith.data.logging</td>
<td>0.9.44</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>esri-geometry-api</td>
<td>2.2.2</td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>lodash</td>
<td>4.17.21</td>
<td>4.17.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
<tr>
<td>colors.js</td>
<td>1.4.0</td>
<td>1.4.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available under</td>
</tr>
<tr>
<td></td>
<td></td>
<td>license</td>
</tr>
</tbody>
</table>
1.2485.1 Available under license
1.2486 remove-trailing-separator 1.1.0
   1.2486.1 Available under license
1.2487 zlib 1.2.8
   1.2487.1 Available under license
1.2488 listenablefuture 1.0
   1.2488.1 Available under license
1.2489 restore-cursor 2.0.0 2.0.0
   1.2489.1 Available under license
1.2490 @babel/plugin-bugfix-v8-spread-parameters-in-optional-chaining 7.16.0
   1.2490.1 Available under license
1.2491 path-root-regex 0.1.2 0.1.2
   1.2491.1 Available under license
1.2492 imurmurhash 0.1.4
   1.2492.1 Available under license
1.2493 htrace-core 3.2.0-incubating
   1.2493.1 Available under license
1.2494 oas-schema-walker 1.1.5 1.1.5
   1.2494.1 Available under license
1.2495 use 3.1.1 3.1.1
   1.2495.1 Available under license
1.2496 inquirer 6.2.2 6.2.2
   1.2496.1 Available under license
1.2497 @babel/types 7.16.0
   1.2497.1 Available under license
1.2498 is-binary-path 2.1.0
   1.2498.1 Available under license
1.2499 jnr-posix 3.0.44
   1.2499.1 Available under license
1.2500 read 1.0.7 1.0.7
   1.2500.1 Available under license
1.2501 utils-merge 1.0.1 1.0.1
   1.2501.1 Available under license
1.2502 is-primitive 2.0.0
   1.2502.1 Available under license
1.2503 @babel/generator 7.16.0
   1.2503.1 Available under license
1.2504 ono 4.0.11 4.0.11
   1.2504.1 Available under license
1.2505 callsite 1.0.0 1.0.0
   1.2505.1 Available under license
1.2506 xregexp 4.0.0 4.0.0
   1.2506.1 Available under license
1.2507 range-parser 1.2.0 1.2.0
   1.2507.1 Available under license
1.2508 guava 11.0.2
   1.2508.1 Available under license
1.2509 is-ip 1.0.0
   1.2509.1 Available under license
1.2510 object.getownpropertydescriptors 2.1.2
   1.2510.1 Available under license
1.2511 is-buffer 2.0.5
   1.2511.1 Available under license
1.2512 @babel/plugin-bugfix-safari-id-destructuring-collision-in-function-expression 7.16.2
   1.2512.1 Available under license
1.2513 optionator 0.8.2
   1.2513.1 Available under license
1.2514 passport-local 1.0.0 1.0.0
   1.2514.1 Available under license
1.2515 httpcore5 5.0.2
   1.2515.1 Available under license
1.2516 netty-handler 4.0.48.Final
   1.2516.1 Available under license
1.2517 es6-iterator 2.0.3
   1.2517.1 Available under license
1.2518 codemirror 5.59.2 5.59.2
   1.2518.1 Available under license
1.2519 generate-object-property 1.2.0
   1.2519.1 Available under license
1.2520 batik-slideshow 1.13
   1.2520.1 Available under license
1.2521 babel-plugin-syntax-optional-chaining 7.8.3
   1.2521.1 Available under license
1.2522 is-binary-path 1.0.1
   1.2522.1 Available under license
1.2523 escape-goat 2.1.1
   1.2523.1 Available under license
1.2524 regenerate 1.4.0 1.4.0
   1.2524.1 Available under license
1.2525 perfmark-api 0.19.0
   1.2525.1 Available under license
1.2526 color-name 1.1.1
   1.2526.1 Available under license
1.2527 log4j-over-slf4j 1.7.26
1.2528 mail 1.4.7
   1.2528.1 Available under license
1.2529 to-regex 3.0.2 3.0.2
   1.2529.1 Available under license
1.2530 jetty-http 9.4.41.v20210516
   1.2530.1 Available under license
1.2531 babel-plugin-transform-es2015-modules-umd 6.24.1
   1.2531.1 Available under license
1.2532 string-width 3.1.0 3.1.0
   1.2532.1 Available under license
1.2533 netty-handler-proxy 4.1.42.Final
   1.2533.1 Available under license
1.2534 core-js 2.6.12 2.6.12
   1.2534.1 Available under license
1.2535 regjspARSER 0.1.5
   1.2535.1 Available under license
1.2536 ansi-styles 3.2.1 3.2.1
   1.2536.1 Available under license
1.2537 release-zalgo 1.0.0
   1.2537.1 Available under license
1.2538 @babel/plugin-transform-block-scoping 7.16.0
   1.2538.1 Available under license
1.2539 async 1.5.2 1.5.2
   1.2539.1 Available under license
1.2540 commons-math 2.1
   1.2540.1 Available under license
1.2541 type-check 0.3.2
   1.2541.1 Available under license
1.2542 babel-helper-builder-binary-assignment-operator-visitor 6.24.1
   1.2542.1 Available under license
1.2543 annotations 16.0.3
   1.2543.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>batik-parser</td>
<td>1.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>mitt</td>
<td>1.1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>js-string-escape</td>
<td>1.0.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-messages</td>
<td>6.23.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>opentest4j</td>
<td>1.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>babel-helper-call-delegate</td>
<td>6.24.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>netty-resolver</td>
<td>4.1.42.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>aws-java-sdk-s3</td>
<td>1.11.704</td>
<td>Available under license</td>
</tr>
<tr>
<td>gson</td>
<td>2.8.6</td>
<td>Available under license</td>
</tr>
<tr>
<td>dropwizard-metrics-hadoop-metrics2-reporter</td>
<td>0.1.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>parse-duration</td>
<td>0.1.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>buffer-crc32</td>
<td>0.2.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>platform</td>
<td>1.3.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>object.assign</td>
<td>4.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>alsa</td>
<td>1.1.0-0ubuntu1</td>
<td>Available under license</td>
</tr>
<tr>
<td>native-promise-only</td>
<td>0.8.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>bytes</td>
<td>2.3.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>aws4</td>
<td>1.11.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/highlight</td>
<td>7.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>@babel/helper-simple-access</td>
<td>7.16.0</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.2563.1 Available under license

1.2564 lodash.flattendeep 4.4.0
1.2564.1 Available under license

1.2565 joni 2.1.31
1.2565.1 Available under license

1.2566 proxy-from-env 1.1.0 1.1.0
1.2566.1 Available under license

1.2567 xml-apis-ext 1.3.04
1.2567.1 Available under license

1.2568 esprima 4.0.1 4.0.1
1.2568.1 Available under license

1.2569 parquet-encoding 1.10.0
1.2569.1 Available under license

1.2570 js-base64 2.6.4 2.6.4
1.2570.1 Available under license

1.2571 supports-color 2.0.0
1.2571.1 Available under license

1.2572 guava 30.1-jre
1.2572.1 Available under license

1.2573 audience-annotations 0.5.0
1.2573.1 Available under license

1.2574 latest-version 5.1.0
1.2574.1 Available under license

1.2575 json5 0.5.1 0.5.1
1.2575.1 Available under license

1.2576 jackson-dataformat-yaml 2.4.2
1.2576.1 Available under license

1.2577 statuses 1.5.0 1.5.0
1.2577.1 Available under license

1.2578 jetty-servlets 9.4.41.v20210516
1.2578.1 Available under license

1.2579 safe-buffer 5.1.1 5.1.1
1.2579.1 Available under license

1.2580 protobuf-java 3.13.0
1.2580.1 Available under license

1.2581 commons-pool 2.6.1

1.2582 jetty-continuation 9.3.27.v20190418
1.2582.1 Available under license

1.2583 estraverse 4.3.0 4.3.0
1.2583.1 Available under license
1.2584 h2 1.4.197
1.2585 lodash.debounce 4.0.8 4.0.8
   1.2585.1 Available under license
1.2586 babel-helper-get-function-arity 6.24.1 6.24.1
   1.2586.1 Available under license
1.2587 jsch 0.1.55
1.2588 @babel/plugin-transform-duplicate-keys 7.16.0
   1.2588.1 Available under license
1.2589 yup 0.29.3 0.29.3
   1.2589.1 Available under license
1.2590 babel-node 7.16.0
   1.2590.1 Available under license
1.2591 glob 7.1.3
   1.2591.1 Available under license
1.2592 @babel/helper-get-function-arity 7.10.4
   1.2592.1 Available under license
1.2593 openjpeg 1.6.35
   1.2593.1 Available under license
1.2594 serve-static 1.10.3 1.10.3
   1.2594.1 Available under license
1.2595 passport-strategy 1.0.0 1.0.0
   1.2595.1 Available under license
1.2596 jszip 3.5.0 3.5.0
   1.2596.1 Available under license
1.2597 babel-plugin-transform-es2015-classes 6.24.1
   1.2597.1 Available under license
1.2598 asm 3.2
   1.2598.1 Available under license
1.2599 preserve 0.2.0
   1.2599.1 Available under license
1.2600 @babel/plugin-transform-modules-umd 7.16.0
   1.2600.1 Available under license
1.2601 worker-loader 1.1.1 1.1.1
   1.2601.1 Available under license
1.2602 babel-plugin-transform-es2015-duplicate-keys 6.24.1
   1.2602.1 Available under license
1.2603 grpc-protobuf-lite 1.27.1
   1.2603.1 Available under license
1.2604 csv-parse 4.8.2 4.8.2
   1.2604.1 Available under license
1.2605 internal-slot 1.0.3
   1.2605.1 Available under license
1.2606 append-field 0.1.0 0.1.0
   1.2606.1 Available under license
1.2607 avatica 1.17.0
   1.2607.1 Available under license
1.2608 range-parser 1.0.3 1.0.3
   1.2608.1 Available under license
1.2609 babel-plugin-proposal-optional-chaining 7.16.0
   1.2609.1 Available under license
1.2610 source-map 0.5.7
   1.2610.1 Available under license
1.2611 memory-pager 1.5.0 1.5.0
   1.2611.1 Available under license
1.2612 eslint-import-resolver-node 0.3.6
   1.2612.1 Available under license
1.2613 @babel/highlight 7.10.4
   1.2613.1 Available under license
1.2614 eyes 0.1.8 0.1.8
   1.2614.1 Available under license
1.2615 app-root-path 2.0.1 2.0.1
   1.2615.1 Available under license
1.2616 kotlin 1.3.61
   1.2616.1 Available under license
1.2617 lodash-padstart 4.6.1 4.6.1
   1.2617.1 Available under license
1.2618 base64id 1.0.0 1.0.0
   1.2618.1 Available under license
1.2619 get-stream 5.1.0 5.1.0
   1.2619.1 Available under license
1.2620 @babel/plugin-transform-new-target 7.16.0
   1.2620.1 Available under license
1.2621 locate-path 2.0.0
   1.2621.1 Available under license
1.2622 hbase-protocol 2.4.2
   1.2622.1 Available under license
1.2623 socket.io-parser 3.3.1 3.3.1
1.2623.1 Available under license
1.2624 global-modules 1.0.0 1.0.0
  1.2624.1 Available under license
1.2625 commons-dbcp 2.6.0
  1.2625.1 Available under license
1.2626 hadoop 3.2.2
  1.2626.1 Available under license
1.2627 gensync 1.0.0-beta.2
  1.2627.1 Available under license
1.2628 generate-function 2.0.0 2.0.0
  1.2628.1 Available under license
1.2629 jline-reader 3.12.1
  1.2629.1 Available under license
1.2630 wrap-ansi 6.2.0 6.2.0
  1.2630.1 Available under license
1.2631 odata-parser 1.4.1 1.4.1
  1.2631.1 Available under license
1.2632 expat 2.1.0
  1.2632.1 Available under license
1.2633 getpass 0.1.7 0.1.7
  1.2633.1 Available under license
1.2634 types-color-name 1.1.1
  1.2634.1 Available under license
1.2635 kite-data-core 1.1.0
  1.2635.1 Available under license
1.2636 snake-yaml 1.29
  1.2636.1 Available under license
1.2637 camelcase 5.0.0 5.0.0
  1.2637.1 Available under license
1.2638 jackson-databind 2.11.4
  1.2638.1 Available under license
1.2639 babel-plugin-transform-exponentiation-operator 6.24.1 6.24.1
  1.2639.1 Available under license
1.2640 mongodb-driver-sync 4.2.3
  1.2640.1 Available under license
1.2641 graceful-fs 4.2.4 4.2.4
  1.2641.1 Available under license
1.2642 okhttp 4.8.0
  1.2642.1 Available under license
1.2643 https-proxy-agent 3.0.1 3.0.1
   1.2643.1 Available under license
1.2644 twill-zookeeper 0.6.0-incubating
   1.2644.1 Available under license
1.2645 jsreport-templates 2.4.0 2.4.0
   1.2645.1 Available under license
1.2646 libpciaccess 0.13.4 1
   1.2646.1 Available under license
1.2647 run-async 2.4.1 2.4.1
   1.2647.1 Available under license
1.2648 expand-range 1.8.2
   1.2648.1 Available under license
1.2649 @babel/helper-replace-supers 7.16.0
   1.2649.1 Available under license
1.2650 browserslist 3.2.8 3.2.8
   1.2650.1 Available under license
1.2651 hasha 5.2.2
   1.2651.1 Available under license
1.2652 registry-url 5.1.0
   1.2652.1 Available under license
1.2653 validatejs 0.12.0 0.12.0
   1.2653.1 Available under license
1.2654 streams 2.7.0
1.2655 form-data 2.5.1 2.5.1
   1.2655.1 Available under license
1.2656 @babel/helper-validator-identifier 7.10.4 7.10.4
   1.2656.1 Available under license
1.2657 component-emitter 1.2.1 1.2.1
   1.2657.1 Available under license
1.2658 array-unique 0.3.2
   1.2658.1 Available under license
1.2659 y18n 3.2.1 3.2.1
   1.2659.1 Available under license
1.2660 @babel/plugin-proposal-unicode-property-regex 7.16.0
   1.2660.1 Available under license
1.2661 lodash-flatten 4.4.0 4.4.0
   1.2661.1 Available under license
1.2662 sqlite-jdbc 3.30.1
   1.2662.1 Available under license
1.2663 hadoop-auth 3.2.2
  1.2663.1 Available under license
1.2664 lucene-queries 7.4.0
  1.2664.1 Available under license
1.2665 drill-jdbc-all 1.18.0
  1.2665.1 Available under license
1.2666 @babel/helper-get-function-arity 7.16.0
  1.2666.1 Available under license
1.2667 apr 1.6.5
1.2668 passport 0.4.0 0.4.0
  1.2668.1 Available under license
1.2669 jline-terminal 3.12.1
  1.2669.1 Available under license
1.2670 regexpu-core 2.0.0
  1.2670.1 Available under license
1.2671 onetime 2.0.1 2.0.1
  1.2671.1 Available under license
1.2672 kafka-metadata 2.8.0
  1.2672.1 Available under license
1.2673 i 0.3.6 0.3.6
  1.2673.1 Available under license
1.2674 jsprim 1.4.1 1.4.1
  1.2674.1 Available under license
1.2675 @babel/plugin-transform-for-of 7.16.0
  1.2675.1 Available under license
1.2676 mimic-response 1.0.1
  1.2676.1 Available under license
1.2677 should-util 1.0.1 1.0.1
  1.2677.1 Available under license
1.2678 yargs 3.32.0 3.32.0
  1.2678.1 Available under license
1.2679 jcodings 1.0.55
  1.2679.1 Available under license
1.2680 url-parse 1.4.7 1.4.7
  1.2680.1 Available under license
1.2681 asm-util 7.3.1
  1.2681.1 Available under license
1.2682 test-exclude 6.0.0
  1.2682.1 Available under license
1.2683 get-package-type 0.1.0
  1.2683.1 Available under license

1.2684 pako 0.2.9 0.2.9
  1.2684.1 Available under license

1.2685 async 2.1.2 2.1.2
  1.2685.1 Available under license

1.2686 babel-template 6.26.0 6.26.0
  1.2686.1 Available under license

1.2687 on-headers 1.0.2 1.0.2
  1.2687.1 Available under license

1.2688 pascalcase 0.1.1 0.1.1
  1.2688.1 Available under license

1.2689 kind-of 4.0.0 4.0.0
  1.2689.1 Available under license

1.2690 undefsafe 2.0.5
  1.2690.1 Available under license

1.2691 touch 3.1.0
  1.2691.1 Available under license

1.2692 strip-ansi 5.2.0 5.2.0
  1.2692.1 Available under license

1.2693 jnr-posix 3.0.41
  1.2693.1 Available under license

1.2694 cli-cursor 1.0.2
  1.2694.1 Available under license

1.2695 base64-js 0.0.2 0.0.2
  1.2695.1 Available under license

1.2696 isexe 2.0.0
  1.2696.1 Available under license

1.2697 d-conf 0.24.0-2
  1.2697.1 Available under license

1.2698 libx11 1.6.3-1ubuntu2.1
  1.2698.1 Available under license

1.2699 log4j-api 2.14.0
  1.2699.1 Available under license

1.2700 revalidator 0.1.8 0.1.8
  1.2700.1 Available under license

1.2701 caffeine 2.7.0

1.2702 esquery 1.3.1
  1.2702.1 Available under license
<table>
<thead>
<tr>
<th>Package Name</th>
<th>Version</th>
<th>License Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>regjsparser</td>
<td>0.7.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>commons-beanutils-core</td>
<td>1.8.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>p-locate</td>
<td>4.1.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>tinycolor2</td>
<td>1.4.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>esprima</td>
<td>1.2.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>is-string</td>
<td>1.0.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>chai</td>
<td>4.3.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>jetty-security</td>
<td>9.3.27.v20190418</td>
<td>Available under license</td>
</tr>
<tr>
<td>decode-uri-component</td>
<td>0.2.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>grpc-protobuf</td>
<td>1.27.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>jsreport-core</td>
<td>2.10.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>component-inherit</td>
<td>0.0.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>concat-stream</td>
<td>1.6.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>braces</td>
<td>3.0.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>p-locate 3.0.0</td>
<td>3.0.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>chokidar</td>
<td>1.7.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>accepts</td>
<td>1.2.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>readable-stream</td>
<td>2.3.7</td>
<td>Available under license</td>
</tr>
<tr>
<td>rechoir</td>
<td>0.6.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>jgrapht-core</td>
<td>1.3.0</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.2723 superagent 3.8.3 3.8.3
  1.2723.1 Available under license
1.2724 apache-log4j 1.2.17
  1.2724.1 Available under license
1.2725 j2objc-annotations 1.3
1.2726 drill-shaded-guava 28.2-jre
1.2727 antlr-runtime 4.9
  1.2727.1 Available under license
1.2728 archiver-utils 2.1.0 2.1.0
  1.2728.1 Available under license
1.2729 node-fetch 2.6.1 2.6.1
  1.2729.1 Available under license
1.2730 metrics-health-checks 4.1.0
1.2731 engine.io 1.8.5 1.8.5
  1.2731.1 Available under license
1.2732 libxext 1.3.3-1
  1.2732.1 Available under license
1.2733 mongodb 3.6.6 3.6.6
  1.2733.1 Available under license
1.2734 bluebird 3.4.7 3.4.7
  1.2734.1 Available under license
1.2735 jersey-container-jetty-servlet 2.25.1
  1.2735.1 Available under license
1.2736 debug 4.2.0 4.2.0
  1.2736.1 Available under license
1.2737 electron-to-chromium 1.3.55 1.3.55
  1.2737.1 Available under license
1.2738 @babel/plugin-transform-computed-properties 7.16.0
  1.2738.1 Available under license
1.2739 pcre 8.38-3.1
  1.2739.1 Available under license
1.2740 kotlin 1.3.72
  1.2740.1 Available under license
1.2741 picocolors 1.0.0
  1.2741.1 Available under license
1.2742 netty-codec-http 4.1.42.Final
  1.2742.1 Available under license
1.2743 asm 5.2
1.2744 jaxb-core 2.3.1
1.2744.1 Available under license

1.2745 twill-core 0.6.0-incubating
   1.2745.1 Available under license

1.2746 velocity 2.0
   1.2746.1 Available under license

1.2747 pathval 1.1.1
   1.2747.1 Available under license

1.2748 @babel/plugin-transform-dotall-regex 7.16.0
   1.2748.1 Available under license

1.2749 jackson-dataformat-csv 2.10.5
   1.2749.1 Available under license

1.2750 junit-platform-commons 1.7.0
   1.2750.1 Available under license

1.2751 parquet-hadoop 1.8.1
   1.2751.1 Available under license

1.2752 jackson-module-afterburner 2.12.1
   1.2752.1 Available under license

1.2753 ansi-colors 0.2.1 0.2.1
   1.2753.1 Available under license

1.2754 babel-plugin-transform-es2015-block-scoping 6.26.0
   1.2754.1 Available under license

1.2755 opentracing-util 0.31.0
   1.2755.1 Available under license

1.2756 css-select 1.2.0 1.2.0
   1.2756.1 Available under license

1.2757 @babel/code-frame 7.16.0
   1.2757.1 Available under license

1.2758 to-buffer 1.1.1 1.1.1
   1.2758.1 Available under license

1.2759 safe-json-stringify 1.2.0 1.2.0
   1.2759.1 Available under license

1.2760 jaeger-core 1.5.0
   1.2760.1 Available under license

   1.2761.1 Available under license

1.2762 to-fast-properties 1.0.3 1.0.3
   1.2762.1 Available under license

1.2763 is-array 0.0.1 0.0.1
   1.2763.1 Available under license
1.2764 to-fast-properties 2.0.0
   1.2764.1 Available under license

1.2765 object-inspect 1.10.3 1.10.3
   1.2765.1 Available under license

1.2766 async 1.4.1
   1.2766.1 Available under license

1.2767 jetty-http 9.3.27.v20190418
   1.2767.1 Available under license

1.2768 parseqs 0.0.6 0.0.6
   1.2768.1 Available under license

1.2769 package-json 6.5.0
   1.2769.1 Available under license

1.2770 babel-helper-function-name 6.24.1 6.24.1
   1.2770.1 Available under license

1.2771 acorn 2.7.0
   1.2771.1 Available under license

1.2772 resolve-url 0.2.1 0.2.1
   1.2772.1 Available under license

1.2773 strip-ansi 5.2.0
   1.2773.1 Available under license

1.2774 requires-port 1.0.0 1.0.0
   1.2774.1 Available under license

1.2775 password-hash 1.2.2 1.2.2
   1.2775.1 Available under license

1.2776 websocket-api 9.3.27.v20190418
1.2777 glob 7.1.6 7.1.6
   1.2777.1 Available under license

1.2778 commons-email 1.5
   1.2778.1 Available under license

1.2779 qs 6.5.1 6.5.1
   1.2779.1 Available under license

1.2780 babel-types 6.26.0
   1.2780.1 Available under license

1.2781 mime 2.4.4 2.4.4
   1.2781.1 Available under license

1.2782 org.jacoco.agent.rt 0.7.9
   1.2782.1 Available under license

1.2783 encodeurl 1.0.2 1.0.2
   1.2783.1 Available under license
1.2784 fn-name 3.0.0 3.0.0
   1.2784.1 Available under license
1.2785 avahi 0.6.32~rc+dfsg-1ubuntu2.3
   1.2785.1 Available under license
1.2786 jheaps 0.9
   1.2786.1 Available under license
1.2787 google-http-client 1.20.0
1.2788 jersey-servlet 1.19
1.2789 pkg-dir 4.2.0
   1.2789.1 Available under license
1.2790 sprintf-js 1.1.0 1.1.0
   1.2790.1 Available under license
1.2791 babel-plugin-transform-es2015-template-literals 6.22.0 6.22.0
   1.2791.1 Available under license
1.2792 json-buffer 3.0.0
   1.2792.1 Available under license
1.2793 slf4j 1.7.26
   1.2793.1 Available under license
1.2794 shebang-regex 1.0.0 1.0.0
   1.2794.1 Available under license
1.2795 snapdragon-node 2.1.1 2.1.1
   1.2795.1 Available under license
1.2796 log4j-slf4j-impl 2.14.0
   1.2796.1 Available under license
1.2797 debug 3.2.6 3.2.6
   1.2797.1 Available under license
1.2798 electron-to-chromium 1.3.690
   1.2798.1 Available under license
1.2799 papa-parse 5.1.1 5.1.1
   1.2799.1 Available under license
1.2800 regenerate-unicode-properties 9.0.0
   1.2800.1 Available under license
1.2801 merge-descriptors 1.0.1 1.0.1
   1.2801.1 Available under license
1.2802 finalhandler 0.4.1 0.4.1
   1.2802.1 Available under license
1.2803 handlebars 4.7.6 4.7.6
   1.2803.1 Available under license
1.2804 get-symbol-description 1.0.0
1.2804.1 Available under license
1.2805 common-prefix 1.1.0 1.1.0
   1.2805.1 Available under license
1.2806 jsreport-import-export 2.1.1 2.1.1
   1.2806.1 Available under license
1.2807 babel-plugin-transform-object-rest-spread 6.26.0
   1.2807.1 Available under license
1.2808 lodash-filter 4.6.0 4.6.0
   1.2808.1 Available under license
1.2809 parquet-format 2.4.0
   1.2809.1 Available under license
1.2810 es6-promise 4.2.8 4.2.8
   1.2810.1 Available under license
1.2811 babel-helper-explode-class 6.24.1
   1.2811.1 Available under license
1.2812 is-finite 1.0.2 1.0.2
   1.2812.1 Available under license
1.2813 parquet-common 1.10.0
   1.2813.1 Available under license
1.2814 request-promise 4.2.6 4.2.6
   1.2814.1 Available under license
1.2815 es6-promise 3.2.1 3.2.1
   1.2815.1 Available under license
1.2816 esrecurse 4.3.0
   1.2816.1 Available under license
1.2817 toposort 2.0.2 2.0.2
   1.2817.1 Available under license
1.2818 velocity-engine-core 2.0
   1.2818.1 Available under license
1.2819 avro 1.7.6
   1.2819.1 Available under license
1.2820 istanbul-lib-coverage 3.0.0
   1.2820.1 Available under license
1.2821 sqlline 1.9.0
   1.2821.1 Available under license
1.2822 clean-stack 2.2.0
   1.2822.1 Available under license
1.2823 tunnel-agent 0.6.0 0.6.0
   1.2823.1 Available under license
1.2824 global-prefix 1.0.2 1.0.2
   1.2824.1 Available under license
1.2825 is-number 3.0.0
   1.2825.1 Available under license
1.2826 jsreport-exceljs 3.7.1 3.7.1
   1.2826.1 Available under license
1.2827 @babel/plugin-transform-destructuring 7.16.0
   1.2827.1 Available under license
1.2828 buffer-more-ints 0.0.2 0.0.2
   1.2828.1 Available under license
1.2829 object.omit 2.0.1
   1.2829.1 Available under license
1.2830 protostuff 1.7.1
   1.2830.1 Available under license
1.2831 aws-sign2 0.7.0 0.7.0
   1.2831.1 Available under license
1.2832 silent-spawn 0.4.0 0.4.0
   1.2832.1 Available under license
1.2833 is-data-descriptor 0.1.4
   1.2833.1 Available under license
1.2834 hbase-metrics 2.4.2
   1.2834.1 Available under license
1.2835 xmldom 0.1.27 0.1.27
   1.2835.1 Available under license
1.2836 has-flag 3.0.0
   1.2836.1 Available under license
1.2837 profiler 1.0.2
1.2838 core-js 1.2.7 1.2.7
   1.2838.1 Available under license
1.2839 dnsjava 2.1.7
   1.2839.1 Available under license
1.2840 parquet-jackson 1.8.1
   1.2840.1 Available under license
1.2841 bouncy-castle 1.57
   1.2841.1 Available under license
1.2842 hsts 1.0.0 1.0.0
   1.2842.1 Available under license
1.2843 de.huxhorn.sulky.formatting 0.9.17
   1.2843.1 Available under license
1.2844 jaxb-api 2.3.1
   1.2844.1 Available under license
1.2845 should-equal 2.0.0 2.0.0
   1.2845.1 Available under license
1.2846 picomatch 2.2.2 2.2.2
   1.2846.1 Available under license
1.2847 has-color 0.1.7 0.1.7
   1.2847.1 Available under license
1.2848 lodash-merge 4.6.2 4.6.2
   1.2848.1 Available under license
1.2849 @babel/plugin-transform-classes 7.16.0
   1.2849.1 Available under license
1.2850 groovyjsr 3.0.8
   1.2850.1 Available under license
1.2851 ajv-keywords 3.4.1 3.4.1
   1.2851.1 Available under license
1.2852 redux 3.7.2 3.7.2
   1.2852.1 Available under license
1.2853 puppeteer 2.0.0 2.0.0
   1.2853.1 Available under license
1.2854 pinkie-promise 2.0.1 2.0.1
   1.2854.1 Available under license
1.2855 ajv-keywords 3.5.2 3.5.2
   1.2855.1 Available under license
1.2856 akka 2.5.32
   1.2856.1 Available under license
1.2857 libjpeg-turbo 1.4.2-0ubuntu3.3
   1.2857.1 Available under license
1.2858 union-value 1.0.1 1.0.1
   1.2858.1 Available under license
1.2859 redux-thunk 2.3.0 2.3.0
   1.2859.1 Available under license
1.2860 cheerio 0.22.0 0.22.0
   1.2860.1 Available under license
1.2861 regexp-quote 0.0.0 0.0.0
   1.2861.1 Available under license
1.2862 map-cache 0.2.2 0.2.2
   1.2862.1 Available under license
1.2863 swagger-schema-official 2.0.0-bab6bed 2.0.0-bab6bed
1.2863.1 Available under license
1.2864 netty-codec 4.1.59.Final
  1.2864.1 Available under license
1.2865 scala-reflect 2.13.5
  1.2865.1 Available under license
1.2866 gson 2.2.4
1.2867 hamcrest 1.3
  1.2867.1 Available under license
1.2868 nssocket 0.6.0 0.6.0
  1.2868.1 Available under license
1.2869 jetty-util-ajax 9.4.41.v20210516
  1.2869.1 Available under license
1.2870 wide-align 1.1.3
  1.2870.1 Available under license
1.2871 camelcase 5.3.1 5.3.1
  1.2871.1 Available under license
1.2872 concat-stream 2.0.0 2.0.0
  1.2872.1 Available under license
1.2873 readable-stream 1.1.14 1.1.14
  1.2873.1 Available under license
1.2874 kerb-simplekdc 1.0.0
  1.2874.1 Available under license
1.2875 tough-cookie 2.3.4 2.3.4
  1.2875.1 Available under license
1.2876 jersey-guice 1.19
1.2877 ws 6.2.2
  1.2877.1 Available under license
1.2878 mime 2.6.0
  1.2878.1 Available under license
1.2879 nan 2.10.0 2.10.0
  1.2879.1 Available under license
1.2880 har-validator 2.1.3 2.1.3
  1.2880.1 Available under license
1.2881 fs-writefile-promise 1.0.3 1.0.3
  1.2881.1 Available under license
1.2882 fs-readfile-promise 2.0.1 2.0.1
  1.2882.1 Available under license
1.2883 expand-tilde 2.0.2 2.0.2
  1.2883.1 Available under license
  1.2884.1 Available under license
1.2885 escape-string-regexp 1.0.5 1.0.5
  1.2885.1 Available under license
1.2886 bean-validation-api 2.0.1.Final
  1.2886.1 Available under license
1.2887 co 4.6.0 4.6.0
  1.2887.1 Available under license
1.2888 bson4jackson 2.7.0
  1.2888.1 Available under license
1.2889 freemarker 2.3.28
  1.2889.1 Available under license
1.2890 ansi-regex 2.1.1 2.1.1
  1.2890.1 Available under license
1.2891 body-parser 1.18.3 1.18.3
  1.2891.1 Available under license
1.2892 reflections 0.9.10
  1.2892.1 Available under license
1.2893 slash 1.0.0
  1.2893.1 Available under license
1.2894 should.js 13.2.3 13.2.3
  1.2894.1 Available under license
1.2895 babel-plugin-transform-regenerator 6.26.0
  1.2895.1 Available under license
1.2896 once 1.4.0 1.4.0
  1.2896.1 Available under license
1.2897 sprintf-js 1.0.3 1.0.3
  1.2897.1 Available under license
1.2898 better-assert 1.0.2 1.0.2
  1.2898.1 Available under license
1.2899 mute-stream 0.0.7 0.0.7
  1.2899.1 Available under license
1.2900 bytes 3.0.0 3.0.0
  1.2900.1 Available under license
1.2901 @exodus/schemasafe 1.0.0-rc.3 1.0.0-rc.3
  1.2901.1 Available under license
1.2902 object.assign 4.1.2
  1.2902.1 Available under license
1.2903 jackson-datatype-jdk8 2.10.5
<table>
<thead>
<tr>
<th>Version</th>
<th>Package Name</th>
<th>Version</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2903.1</td>
<td>Available under license</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.2904</td>
<td>google-discovery-to-swagger 2.1.0 2.1.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2905</td>
<td>librsvg 2.40.13-3</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2906</td>
<td>pupa 2.1.1</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2907</td>
<td>is-property 1.0.2</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2908</td>
<td>@babel/plugin-proposal-numeric-separator 7.16.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2909</td>
<td>process-nextick-args 1.0.7 1.0.7</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2910</td>
<td>hbase-logging 2.4.2</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2911</td>
<td>nopt 1.0.10 1.0.10</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2912</td>
<td>@babel/plugin-transform-unicode-escapes 7.16.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2913</td>
<td>deep-is 0.1.3 0.1.3</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2914</td>
<td>drill-format-httpd 1.19.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2915</td>
<td>strip-bom 4.0.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2916</td>
<td>oauth-sign 0.8.2 0.8.2</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2917</td>
<td>define-property 1.0.0 1.0.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2918</td>
<td>archy 1.0.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2919</td>
<td>vector 1.19.0</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2920</td>
<td>netty-transport-native-unix-common 4.1.59.Final</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2921</td>
<td>protostuff-json 1.7.1</td>
<td></td>
<td>Available under license</td>
</tr>
<tr>
<td>1.2922</td>
<td>summernote 0.8.11 0.8.11</td>
<td></td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.2923 jackson-dataformat-yaml 2.9.8
   1.2923.1 Available under license
1.2924 hsqldb-jdbc 1.8.0.10
   1.2924.1 Available under license
1.2925 has-symbols 1.0.2
   1.2925.1 Available under license
1.2926 mkdirp-promise 1.1.0 1.1.0
   1.2926.1 Available under license
1.2927 jsreport-pptx 0.7.0 0.7.0
   1.2927.1 Available under license
1.2928 string-width 2.1.1 2.1.1
   1.2928.1 Available under license
1.2929 drill-storage-cassandra 1.19.0
   1.2929.1 Available under license
1.2930 lodash 4.17.19 4.17.19
   1.2930.1 Available under license
1.2931 curl 7.51.0
   1.2931.1 Available under license
1.2932 javax-el-api 2.2.4
   1.2932.1 Available under license
1.2933 make-dir 2.1.0
   1.2933.1 Available under license
1.2934 joda-time 2.9.9
   1.2934.1 Available under license
1.2935 commander 2.20.3 2.20.3
   1.2935.1 Available under license
1.2936 parquet-encoding 1.8.1
   1.2936.1 Available under license
1.2937 debug 2.3.3 2.3.3
   1.2937.1 Available under license
1.2938 content-disposition 0.5.3 0.5.3
   1.2938.1 Available under license
1.2939 get-stream 4.1.0 4.1.0
   1.2939.1 Available under license
1.2940 p-try 2.2.0 2.2.0
   1.2940.1 Available under license
1.2941 @babel/plugin-proposal-private-property-in-object 7.16.0
   1.2941.1 Available under license
1.2942 zstd 1.4.9
1.2942.1 Available under license
1.2943 amcharts3-animate 1.1.2 1.1.2
   1.2943.1 Available under license
1.2944 should-format 3.0.3 3.0.3
   1.2944.1 Available under license
1.2945 debug 2.2.0 2.2.0
   1.2945.1 Available under license
1.2946 commons-compress 1.14
   1.2946.1 Available under license
1.2947 okio 2.7.0
   1.2947.1 Available under license
1.2948 xmlsec-java 2.2.1
   1.2948.1 Available under license
1.2949 is-number 2.1.0
   1.2949.1 Available under license
1.2950 lodash-es 4.17.10 4.17.10
   1.2950.1 Available under license
1.2951 aop-alliance 1.0
   1.2951.1 Available under license
1.2952 debug 4.1.1 4.1.1
   1.2952.1 Available under license
1.2953 resolve 1.20.0
   1.2953.1 Available under license
1.2954 busboy 0.2.14 0.2.14
   1.2954.1 Available under license
1.2955 setimmediate 1.0.5 1.0.5
   1.2955.1 Available under license
1.2956 array-each 1.0.1 1.0.1
   1.2956.1 Available under license
1.2957 fill-range 7.0.1
   1.2957.1 Available under license
1.2958 accessors-smart 1.2
   1.2958.1 Available under license
1.2959 amcharts3-export 1.4.73 1.4.73
   1.2959.1 Available under license
1.2960 orc-core 1.5.6
   1.2960.1 Available under license
1.2961 cookie-parser 1.4.4 1.4.4
   1.2961.1 Available under license
1.2962 split-string 3.1.0 3.1.0
1.2962.1 Available under license

1.2963 npm-install-package 2.1.0 2.1.0
1.2963.1 Available under license

1.2964 jackson 2.2.3
1.2964.1 Available under license

1.2965 send 0.17.1 0.17.1
1.2965.1 Available under license

1.2966 qdox 1.12.1

1.2967 drill-common 1.19.0
1.2967.1 Available under license

1.2968 jline-terminal-jansi 3.12.1
1.2968.1 Available under license

1.2969 jackson 2.11.4
1.2969.1 Available under license

1.2970 detect-file 1.0.0 1.0.0
1.2970.1 Available under license

1.2971 hikaricp-java7 2.4.13

1.2972 proxy-addr 1.0.10 1.0.10
1.2972.1 Available under license

1.2973 cassandra-driver-core 3.2.0

1.2974 is-absolute-url 2.1.0 2.1.0
1.2974.1 Available under license

1.2975 wordwrap 1.0.0 1.0.0
1.2975.1 Available under license

1.2976 popperjs 1.15.0 1.15.0
1.2976.1 Available under license

1.2977 @babel/plugin-proposal-export-namespace-from 7.16.0
1.2977.1 Available under license

1.2978 is-obj 2.0.0
1.2978.1 Available under license

1.2979 object-assign 2.1.1 2.1.1
1.2979.1 Available under license

1.2980 tslib 1.14.1 1.14.1
1.2980.1 Available under license

1.2981 unicode-canonical-property-names-ecmascript 2.0.0
1.2981.1 Available under license

1.2982 lowercase-keys 1.0.1 1.0.1
1.2982.1 Available under license
1.2983 oas-validator 5.0.5 5.0.5
1.2983.1 Available under license

1.2984 engine.io-client 3.4.4 3.4.4
1.2984.1 Available under license

1.2985 ignite-indexing 2.11.0
1.2985.1 Available under license

1.2986 commons-daemon 1.0.13
1.2986.1 Available under license

1.2987 underscore 1.9.2 1.9.2
1.2987.1 Available under license

1.2988 hive-contrib 3.1.2
1.2988.1 Available under license

1.2989 regenerator-runtime 0.11.1 0.11.1
1.2989.1 Available under license

1.2990 resolve 1.9.0
1.2990.1 Available under license

1.2991 cairo 1.14.6 1
1.2991.1 Available under license

1.2992 htmlparser2 3.10.1 3.10.1
1.2992.1 Available under license

1.2993 request-local 1.0.5 1.0.5
1.2993.1 Available under license

1.2994 babel-plugin-proposal-dynamic-import 7.16.0
1.2994.1 Available under license

1.2995 buffer-alloc-unsafe 1.1.0 1.1.0
1.2995.1 Available under license

1.2996 jdo-api 3.0.1
1.2997 fined 1.2.0 1.2.0
1.2997.1 Available under license

1.2998 swagger-annotations 1.5.0
1.2998.1 Available under license

1.1 simple-odata-server 1.1.1 1.1.1
1.1.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Jan Blaha

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2 script-manager 0.10.2 0.10.2
1.2.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Jan Blaha

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.3 isomorphic-fetch 2.2.1 2.2.1
1.3.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Matt Andrews

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.4 ansi-escapes 3.2.0 3.2.0

1.4.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.5 make-dir 3.1.0

1.5.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.6 cookie-signature 1.0.6 1.0.6

1.6.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.7 jetty-xml 9.4.34.v20201102

1.7.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and
consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both
* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.8 babel-helpers 6.24.1
1.8.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.9 zlib 1.2.3

1.9.1 Available under license:
/* zlib.h -- interface of the 'zlib' general purpose compression library
 version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org           madler@alumni.caltech.edu
*/
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.10 kerby-pkix 1.0.0
1.10.1 Available under license:

Kerby PKIX Project
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.11 commondir 1.0.1

1.11.1 Available under license:
The MIT License

Copyright (c) 2013 James Halliday (mail@substack.net)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.12 kind-of 6.0.3

1.12.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.13 to-object-path 0.3.0 0.3.0

1.13.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.14 lodash.isequal 4.5.0 4.5.0

1.14.1 Available under license:

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.15 batik-svg-dom 1.13
1.15.1 Available under license:
No license file was found, but licenses were detected in source scan.

<!-- .............................................................. -->
<!-- SVG 1.1 Core Attribute Module ................................-->
<!-- file: svg-core-attrib.mod -->

This is SVG, a language for describing two-dimensional graphics in XML.
This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Core Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-core-attrib.mod"

<!ENTITY % SVG.id.attrib "id ID #IMPLIED" >
<!ENTITY % SVG.base.attrib "xml:base %URI.datatype; #IMPLIED" >
<!ENTITY % SVG.lang.attrib "xml:lang %LanguageCode.datatype; #IMPLIED" >
<!ENTITY % SVG.space.attrib "xml:space ( default | preserve ) #IMPLIED" >
<!ENTITY % SVG.Core.extra.attrib "" >
<!ENTITY % SVG.Core.attrib "%SVG.id.attrib;
%SVG.base.attrib;
%SVG.lang.attrib;
%SVG.space.attrib;
%SVG.Core.extra.attrib;" >

<!-- end of svg-core-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-profile.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
PUBLIC "//W3C//ELEMENTS SVG 1.1 Color Profile//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-profile.mod"
```

This module declares markup to provide support for color profile.

```
<!DOCTYPE color-profile SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-profile.mod" >
```

```
<!ENTITY % SVG.color-profile.qname "color-profile" >
```

```
<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
```

```
<!ENTITY % SVG.Profile.class | %SVG.color-profile.qname; %SVG.Profile.extra.class; >
```

```
<!ENTITY % SVG.Profile.attrib "color-profile CDATA #IMPLIED
```
<!-- SVG 1.1 Tiny Attribute Collection

This module defines the set of common attributes that can be present on many SVG elements.

--> 

<!-- module: svg-conditional.mod ....................... --> 

<!ENTITY % ExtensionList.datatype "CDATA" > 
<!ENTITY % FeatureList.datatype "CDATA" > 

<!ENTITY % SVG.Conditional.extra.attrib "" > 
<!ENTITY % SVG.Conditional.attrib 
"requiredFeatures %FeatureList.datatype; #IMPLIED
requiredExtensions %ExtensionList.datatype; #IMPLIED
systemLanguage %LanguageCodes.datatype; #IMPLIED
%SVG.Conditional.extra.attrib;" 
>

<!-- module: svg-style.mod ......................... --> 

<!ENTITY % ClassList.datatype "CDATA" > 
<!ENTITY % StyleSheet.datatype "CDATA" > 

<!ENTITY % SVG.Style.extra.attrib "" > 
<!ENTITY % SVG.Style.attrib 
"style %StyleSheet.datatype; #IMPLIED
class %ClassList.datatype; #IMPLIED
%SVG.Style.extra.attrib;" 
>

<!-- module: svg-basic-text.mod ...................... --> 

<!ENTITY % FontFamilyValue.datatype "CDATA" > 
<!ENTITY % FontSizeValue.datatype "CDATA" > 

<!ENTITY % SVG.Font.extra.attrib "" > 
<!ENTITY % SVG.Font.attrib 
"font-family %FontFamilyValue.datatype; #IMPLIED
font-size %FontSizeValue.datatype; #IMPLIED
font-style ( normal | italic | oblique | inherit ) #IMPLIED
font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
%SVG.Font.extra.attrib;" 
>

<!-- end of svg11-tiny-attribs.mod -->
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-cursor.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Cursor//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-cursor.mod"

This module declares markup to provide support for cursor.

<!-- 'cursor' property/attribute value (e.g., 'crosshair', <uri>) -->
<!ENTITY % CursorValue.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->
<!ENTITY % SVG.cursor.qname "cursor" >

<!-- Attribute Collections (Default) ................... -->
<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Cursor.class .................................. -->
<!ENTITY % SVG.Cursor.extra.class "" >

<!-- SVG.Cursor.class .................................. -->
<!ENTITY % SVG.Cursor.class 
"| %SVG.cursor.qname; %SVG.Cursor.extra.class;"
SVG 1.1 Tiny DTD

This is SVG Tiny, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
Permission to use, copy, modify and distribute the SVG DTD and its accompanying documentation for any purpose and without fee is hereby granted in perpetuity, provided that the above copyright notice and this paragraph appear in all copies. The copyright holders make no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/resources/svg11-tiny.dtd
No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Basic Graphics Attribute Module ............................... -->
file: svg-basic-graphics-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-basic-graphics-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Graphics Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-graphics-attrib.mod"

....................................................................... -->

<!-- Basic Graphics Attribute -->

display, visibility

This module defines the Graphics attribute set.

-->

<!ENTITY % SVG.display.attrib
"display ( inline | block | list-item | run-in | compact | marker |
  table | inline-table | table-row-group | table-header-group |
  table-footer-group | table-row | table-column-group |
  table-column | table-cell | table-caption | none | inherit )
#IMPLIED"
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-framework.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Modular Framework//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-framework.mod"

This module instantiates the modules needed to support the SVG modularization model, including:

+ Datatypes
+ Qualified Name
+ Document Model
+ Attribute Collection

<!ENTITY % svg-datatypes.module "INCLUDE" >
<![%svg-datatypes.module;]
<!ENTITY % svg-datatypes.mod
PUBLIC "//W3C//ENTITIES SVG 1.1 Datatypes//EN"
"svg-datatypes.mod" >
%svg-datatypes.mod;]]>

<!ENTITY % svg-qname.module "INCLUDE" >
<![%svg-qname.module;]
<!ENTITY % svg-qname.mod
PUBLIC "//W3C//ENTITIES SVG 1.1 Qualified Name//EN"
"svg-qname.mod" >
%svg-qname.mod;]]>

<!ENTITY % svg-model.module "INCLUDE" >
<![%svg-model.module;]
<!-- instantiate the Document Model declared in the DTD driver -->
%svg-model.mod;]]>

<!ENTITY % svg-attribs.module "INCLUDE" >
<![%svg-attribs.module;]
<!-- instantiate the Attribute Collection declared in the DTD driver -->
%svg-attribs.mod;]]>

<!-- end of svg-framework.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg-framework.mod
No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->

<!-- SVG 1.1 Structure Module .............................................. -->
<file: svg-structure.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-structure.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "//W3C//ELEMENTS SVG 1.1 Structure//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-structure.mod"

....................................................................... -->

<!-- Structure

svg, g, defs, desc, title, metadata, symbol, use

Open Source Used In DNAC Platform Arica 188
This module declares the major structural elements and their attributes.

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.svg.qname "svg" >
<!ENTITY % SVG.g.qname "g" >
<!ENTITY % SVG.defs.qname "defs" >
<!ENTITY % SVG.desc.qname "desc" >
<!ENTITY % SVG.title.qname "title" >
<!ENTITY % SVG.metadata.qname "metadata" >
<!ENTITY % SVG.symbol.qname "symbol" >
<!ENTITY % SVG.use.qname "use" >

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVGOpacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.DocumentEvents.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Description.class ............................. -->

<!ENTITY % SVG.Description.extra.class "" >
<!ENTITY % SVG.Description.class
"%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname;"
Open Source Used In DNAC Platform Arica 194
The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.
Permission to use, copy, modify and distribute the SVG DTD and its accompanying documentation for any purpose and without fee is hereby granted in perpetuity, provided that the above copyright notice and this paragraph appear in all copies. The copyright holders make no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

dimensional graphics in XML.

Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Structure//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-structure.mod"

<!-- Basic Structure

svg, g, defs, desc, title, metadata, use

This module declares the major structural elements and their attributes. -->

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.svg.qname "svg" >
<!ENTITY % SVG.g.qname "g" >
<!ENTITY % SVG.defs.qname "defs" >
<!ENTITY % SVG.desc.qname "desc" >
<!ENTITY % SVG.title.qname "title" >
<!ENTITY % SVG.metadata.qname "metadata" >
<!ENTITY % SVG.use.qname "use" >

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.DocumentEvents.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Description.class ............................. -->
<!ENTITY % SVG.Description.extra.class "" >
<!ENTITY % SVG.Description.class "%SVG.desc.qname; | %SVG.title.qname; | %SVG.metadata.qname; %SVG.Description.extra.class;" />

<!-- SVG.Use.class ..................................... -->
<!ENTITY % SVG.Use.extra.class "" >
<!ENTITY % SVG.Use.class "| %SVG.use.qname; %SVG.Use.extra.class;" >
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-shape.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Shape//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-shape.mod"

....................................................................... -->

<!-- Shape -->

path, rect, circle, line, ellipse, polyline, polygon

This module declares markup to provide support for graphical shapes.

<!-- a list of points -->
<!ENTITY % Points.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.path.qname "path" >
<!ENTITY % SVG.rect.qname "rect" >
<!ENTITY % SVG.circle.qname "circle" >
<!ENTITY % SVG.line.qname "line" >
<!ENTITY % SVG.ellipse.qname "ellipse" >
<!ENTITY % SVG.polyline.qname "polyline" >
<!ENTITY % SVG.polygon.qname "polygon" >

<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!-- Gradient

linearGradient, radialGradient, stop

This module declares markup to provide support for gradient fill.
--> 

<!-- a <number> or a <percentage> -->
<!ENTITY % NumberOrPercentage.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.linearGradient.qname "linearGradient" >
<!ENTITY % SVG.radialGradient.qname "radialGradient" >
<!ENTITY % SVG.stop.qname "stop" >

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Gradient.class ................................ -->
<!ENTITY % SVG.Gradient.extra.class "" >

<!ENTITY % SVG.Gradient.class
"| %SVG.linearGradient.qname; | %SVG.radialGradient.qname; %SVG.Gradient.extra.class;"
>

<!-- SVG.Gradient.attrib ................................ -->
<!ENTITY % SVG.Gradient.extra.attrib "" >

<!ENTITY % SVG.Gradient.attrib
"stop-color %SVGColor.datatype; #IMPLIED
stop-opacity %OpacityValue.datatype; #IMPLIED %SVG.Gradient.extra.attrib;"
>

<!-- linearGradient: Linear Gradient Element .......... -->
%SVG.Core.attrib;
%SVG.Style.attrib;
%SVG.Color.attrib;
%SVG.Gradient.attrib;
%SVG.XLink.attrib;
%SVG.External.attrib;
cx %Coordinate.datatype; #IMPLIED
cy %Coordinate.datatype; #IMPLIED
r %Length.datatype; #IMPLIED
fx %Coordinate.datatype; #IMPLIED
fy %Coordinate.datatype; #IMPLIED
gradientUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
gradientTransform %TransformList.datatype; #IMPLIED
spreadMethod ( pad | reflect | repeat ) #IMPLIED
>
<!-- end of SVG.radialGradient.attlist -->]]>

<!ENTITY % SVG.stop.extra.content "" >

<!ENTITY % SVG.stop.element "INCLUDE" >
<![%SVG.stop.element;[
<!ENTITY % SVG.stop.content
"( %SVG.animate.qname; | %SVG.set.qname; | %SVG.animateColor.qname;
  %SVG.stop.extra.content; )*"
>
<!ELEMENT %SVG.stop.qname; %SVG.stop.content; >
<!-- end of SVG.stop.element -->]]>

<!ENTITY % SVG.stop.attlist "INCLUDE" >
<![%SVG.stop.attlist;[
<!ATTLIST %SVG.stop.qname;
  %SVG.Core.attrib;
  %SVG.Style.attrib;
  %SVG.Color.attrib;
  %SVG.Gradient.attrib;
  offset %NumberOrPercentage.datatype; #REQUIRED
>
<!-- end of SVG.stop.attlist -->]]>

<!-- end of svg-gradient.mod -->

Found in path(s):
  * /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/resources/svg-gradient.mod

No license file was found, but licenses were detected in source scan.
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-marker.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Marker//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-marker.mod"

--------------------------------------------- -->

<!-- Marker -->

marker

This module declares markup to provide support for marker.

<!-- 'marker' property/attribute value (e.g., 'none', <uri>) -->

<!ENTITY % MarkerValue.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.marker.qname "marker" >

<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!ENTITY % SVG.Marker.class "" >

<!ENTITY % SVG.Marker.extra.class "" >

<!ENTITY % SVG.Marker.class "|
%SVG.marker.qname; %SVG.Marker.extra.class;" >

<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Marker.extra.attrib "" >

<!ENTITY % SVG.Marker.attrib "" >
"marker-start %MarkerValue.datatype; #IMPLIED
marker-mid %MarkerValue.datatype; #IMPLIED
marker-end %MarkerValue.datatype; #IMPLIED
%SVG.Marker.extra.attrib;" >

<!ENTITY % SVG.Presentation.attrib "" >
<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib "" >
"%SVG.Container.attrib;
%SVG.Viewport.attrib;
%SVG.Text.attrib;
%SVG.TextContent.attrib;
%SVG.Font.attrib;
%SVG.Paint.attrib;
%SVG.Color.attrib;
%SVG.Opacity.attrib;
%SVG.Graphics.attrib;
%SVG.Marker.attrib;
%SVG.Profile.attrib;
%SVG.Gradient.attrib;
%SVG.Clip.attrib;
%SVG.Mask.attrib;
%SVG.Filter.attrib;
%SVG.FilterColor.attrib;
%SVG.Cursor.attrib;
flood-color %SVGColor.datatype; #IMPLIED
flood-opacity %OpacityValue.datatype; #IMPLIED
lighting-color %SVGColor.datatype; #IMPLIED
%SVG.Presentation.extra.attrib;"
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-extensibility.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Extensibility//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-extensibility.mod"

................................................................. -->

<!-- Extensibility

foreignObject

This module declares markup to provide support for extensibility. -->

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.foreignObject.qname "foreignObject" >

<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Extensibility.class ......................... -->
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-mask.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

```
PUBLIC "//W3C//ELEMENTS SVG 1.1 Mask//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-mask.mod"
```

This module declares markup to provide support for masking.

```
<!ENTITY % MaskValue.datatype "CDATA" >

<!-- Qualified Names (Default) ..................... -->
```
"%SVG.Container.attrib;
%SVG.Viewport.attrib;
%SVG.Text.attrib;
%SVG.TextContent.attrib;
%SVG.Font.attrib;
%SVG.Paint.attrib;
%SVG.Color.attrib;
%SVG.Opacity.attrib;
%SVG.Graphics.attrib;
%SVG.Marker.attrib;
%SVG.Profile.attrib;
%SVG.Gradient.attrib;
%SVG.Clip.attrib;
%SVG.Mask.attrib;
%SVG.Filter.attrib;
%SVG.FilterColor.attrib;
%SVG.Cursor.attrib;
flood-color %SVGColor.datatype; #IMPLIED
flood-opacity %OpacityValue.datatype; #IMPLIED
lighting-color %SVGColor.datatype; #IMPLIED
%SVG.Presentation.extra.attrib;"
>

<!-- mask: Mask Element ................................ -->
<!ENTITY % SVG.mask.extra.content "" >

<!ENTITY % SVG.mask.element "INCLUDE" >
<![%SVG.mask.element;]
<!ENTITY % SVG.mask.content
"( %SVG.Description.class; | %SVG.Animation.class; %SVG.Structure.class;
  %SVG.Conditional.class; %SVG.Image.class; %SVG.Style.class;
  %SVG.Shape.class; %SVG.Text.class; %SVG.Marker.class; %SVG.Profile.class;
  %SVG.Gradient.class; %SVG.Pattern.class; %SVG.Clip.class;
  %SVG.Mask.class; %SVG.Filter.class; %SVG.Cursor.class;
  %SVG.Hyperlink.class; %SVG.View.class; %SVG.Script.class;
  %SVG.Font.class; %SVG.mask.extra.content; )*"
>
<!ELEMENT %SVG.mask.qname; %SVG.mask.content; >
<!-- end of SVG.mask.element -->]]>

<!ENTITY % SVG.mask.attlist "INCLUDE" >
<![%SVG.mask.attlist;]
<!ATTLIST %SVG.mask.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-model.mod 1733416 2016-03-03 07:07:13Z gadams $
Open Source Used In DNAC Platform Arica 222
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-view.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 View//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-view.mod"

This module declares markup to provide support for view.
<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.view.qname "view">

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.View.class .................................... -->

<!ENTITY % SVG.View.extra.class "" >

<!ENTITY % SVG.View.class
    "| %SVG.view.qname; %SVG.View.extra.class;"
>

<!-- view: View Element ................................ -->

<!ENTITY % SVG.view.element "INCLUDE" >
<![%SVG.view.element;]
<!ENTITY % SVG.view.content
    "( %SVG.Description.class; %SVG.view.extra.content; )*"
>
<!ELEMENT %SVG.view.qname; %SVG.view.content; -->

<!ENTITY % SVG.view.attlist "INCLUDE" >
<![%SVG.view.attlist;]
<!ATTLIST %SVG.view.qname;
    %SVG.Core.attrib;
    %SVG.External.attrib;
    viewBox %ViewBoxSpec.datatype; #IMPLIED
    preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
    zoomAndPan ( disable | magnify ) 'magnify'
    viewTarget CDATA #IMPLIED
>
<!-- end of SVG.view.attlist -->]

<!-- end of svg-view.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/resources/svg-view.mod
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 World Wide Web Consortium,
 * 
 * (Massachusetts Institute of Technology, European Research Consortium for
 * Informatics and Mathematics, Keio University). All Rights Reserved. This
 * work is distributed under the W3C(r) Software License [1] in the hope that
 * it will be useful, but WITHOUT ANY WARRANTY; without even the implied
 * warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
 * 
 * 
 * Modifications:
 * 
 * September 18, 2005
 * Translated IDL to Java.
 * Placed interface in org.apache.batik.dom.svg12 for the time being.
 * Added javadocs.
 */

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg12/SVGGlobal.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg12/Global.java

No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Animation Events Attribute Module ............................. -->
<!-- file: svg-animevents-attrib.mod
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-animevents-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

    PUBLIC "-//W3C//ENTITIES SVG 1.1 Animation Events Attribute//EN"
    SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-animevents-attrib.mod"

....................................................................... -->

<!-- Animation Events Attribute

onbegin, onend, onrepeat, onload

This module defines the AnimationEvents attribute set. --->
<ENTITY % SVG.onbegin.attrib
  "onbegin %Script.datatype; #IMPLIED"
>

<ENTITY % SVG.onend.attrib
  "onend %Script.datatype; #IMPLIED"
>

<ENTITY % SVG.onrepeat.attrib
  "onrepeat %Script.datatype; #IMPLIED"
>

<ENTITY % SVG.onload.attrib
  "onload %Script.datatype; #IMPLIED"
>

<ENTITY % SVG.AnimationEvents.extra.attrib ""
>

<ENTITY % SVG.AnimationEvents.attrib
  "%SVG.onbegin.attrib;
  %SVG.onend.attrib;
  %SVG.onrepeat.attrib;
  %SVG.onload.attrib;
  %SVG.AnimationEvents.extra.attrib;"
>

<!-- end of svg-animevents-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1- jar/org/apache/batik/dom/svg/resources/svg-animevents-attrib.mod

No license file was found, but licenses were detected in source scan.

<!-- ................................................................. -->
<!-- SVG 1.1 Basic Document Model Module ......................... -->
<!-- file: svg11-basic-model.mod

This is SVG Basic, a proper subset of SVG.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-basic-model.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Basic Document Model//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-basic-model.mod"
<!-- SVG 1.1 Basic Document Model

This module describes the groupings of elements that make up common content models for SVG elements.

-->
Public identifier:
PUBLIC "

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik(dom/svg/resources/svg10.dtd
No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Hyperlinking Module ........................................... -->
<!-- file: svg-hyperlink.mod -->

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-hyperlink.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Hyperlinking//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-hyperlink.mod"

..................................................................... -->

<!-- Hyperlinking -->

a

This module declares markup to provide support for hyper linking.

-->

<!-- link to this target -->
<!-- ENTITY % LinkTarget.datatype "NM_TOKEN" -->

<!-- Qualified Names (Default) ......................... -->

<!-- ENTITY % SVG.a.qname "a" -->

<!-- Attribute Collections (Default) ................... -->

<!-- ENTITY % SVG.Core.attrib "" -->
<!-- ENTITY % SVG.Conditional.attrib "" -->
<!-- ENTITY % SVG.Style.attrib "" -->
<!-- ENTITY % SVG.Viewport.attrib "" -->
<!-- ENTITY % SVG.Text.attrib "" -->
<!-- ENTITY % SVG.TextContent.attrib "" -->
<!-- ENTITY % SVG.Font.attrib "" -->
<!-- ENTITY % SVG.Paint.attrib "" -->
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkReplace.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Hyperlink.class ............................... -->
<!ENTITY % SVG.Hyperlink.class "" >

<!-- SVG.Presentation.attrib ........................... -->
<!ENTITY % SVG.Presentation.extra.attrib "" >
<!ENTITY % SVG.Presentation.attrib "%SVG.Container.attrib;
%SVG.Viewport.attrib;
%SVG.Text.attrib;
%SVG.TextContent.attrib;
%SVG.Font.attrib;
%SVG.Paint.attrib;
%SVG.Color.attrib;
%SVG.Opacity.attrib;
%SVG.Graphics.attrib;
%SVG.Marker.attrib;
%SVG.Profile.attrib;
%SVG.Gradient.attrib;
%SVG.Clip.attrib;
%SVG.Mask.attrib;
%SVG.Filter.attrib;
%SVG.FilterColor.attrib;
%SVG.Cursor.attrib;
flood-color %SVGColor.datatype; #IMPLIED
flood-opacity %OpacityValue.datatype; #IMPLIED
lighting-color %SVGColor.datatype; #IMPLIED
%SVG.Presentation.extra.attrib;"
Open Source Used In DNAC Platform Arica 235

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "//W3C//ENTITIES SVG 1.1 Container Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-container-attrib.mod"

....................................................................... -->

<!-- Container Attribute

enable-background

This module defines the Container attribute set.
-->

<!-- 'enable-background' property/attribute value (e.g., 'new', 'accumulate') -->
<!-- ENTITY % EnableBackgroundValue.datatype "CDATA" -->

<!-- ENTITY % SVG.enable-background.attrib

"enable-background %EnableBackgroundValue.datatype; #IMPLIED"
>

<!-- ENTITY % SVG.Container.extra.attrib "" -->

<!-- ENTITY % SVG.Container.attrib

"%SVG.enable-background.attrib;
 %SVG.Container.extra.attrib;"
>

<!-- end of svg-container-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg-container-attrib.mod
No license file was found, but licenses were detected in source scan.

<!-- ................................................................. -->

<!-- SVG 1.1 Clip Module ............................................ -->

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-clip.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "//W3C//ELEMENTS SVG 1.1 Clip//EN"
This module declares markup to provide support for clipping.

`clip-path` property/attribute value (e.g., 'none', `<uri>`) -->

`clip-path` attribute collections (Default) ................... -->

SVG.Clip.class .....................................

SVG.Clip.attrib ...................................

<!ENTITY % SVG.Clip.class "| %SVG.clipPath.qname; %SVG.Clip.extra.class;" >

<!ENTITY % SVG.Clip.attrib "" >

<!ENTITY % SVG.Clip.extra.attrib "" >

<!ENTITY % SVG.Clip.attrib"" >
"clip-path %ClipPathValue.datatype; #IMPLIED
clip-rule %ClipFillRule.datatype; #IMPLIED
%SVG.Clip.extra.attrib;"

<!-- clipPath: Clip Path Element ....................... -->

<!ENTITY % SVG.clipPath.extra.content "" >

<!ENTITY % SVG.clipPath.element "INCLUDE" >
<![%SVG.clipPath.element;[
<!ENTITY % SVG.clipPath.content
"(( %SVG.Description.class; )*, ( %SVG.Animation.class; %SVG.Use.class;
 %SVG.Shape.class; %SVG.Text.class; %SVG.clipPath.extra.content; )*)"
]>
<!ELEMENT %SVG.clipPath.qname; %SVG.clipPath.content; >
<!-- end of SVG.clipPath.element -->]]>

<!ENTITY % SVG.clipPath.attlist "INCLUDE" >
<![%SVG.clipPath.attlist;[
<!ATTLIST %SVG.clipPath.qname;
 %SVG.Core.attrib;
 %SVG.Conditional.attrib;
 %SVG.Style.attrib;
 %SVG.Text.attrib;
 %SVG.TextContent.attrib;
 %SVG.Font.attrib;
 %SVG.Paint.attrib;
 %SVG.Color.attrib;
 %SVG.Opacity.attrib;
 %SVG.Graphics.attrib;
 %SVG.Clip.attrib;
 %SVG.Mask.attrib;
 %SVG.Filter.attrib;
 %SVG.Cursor.attrib;
 %SVG.External.attrib;
 transform %TransformList.datatype; #IMPLIED
 clipPathUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
]>
<!-- end of SVG.clipPath.attlist -->]]>

<!-- end of svg-clip.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
 jar/org/apache/batik/dom/svg/resources/svg-clip.mod
No license file was found, but licenses were detected in source scan.
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/SVGAnimatedPathDataSupport.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/SVGItem.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/TraitAccess.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/SVGOMTransform.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/SVGMotionAnimatableElement.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/ListHandler.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/AbstractSVGPathSegList.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/IdContainer.java
Open Source Used In DNAC Platform Arica 240

* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGPathContext.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg12/SVGMOMWheelEvent.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGPathSegItem.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/LiveAttributeException.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGOMAngle.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/AbstractSVGMatrix.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGOMEvent.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGZoomAndPanSupport.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/AbstractSVGPointList.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/ExtendedTraitAccess.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGSVGContext.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGOMUseShadowRoot.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGDocumentFactory.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGMOMPoint.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/LiveAttributeValue.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/AbstractSVGList.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/AbstractSVGTransform.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGContext.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGMOMMatrix.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/AbstractSVGPreserveAspectRatio.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/SVGMORect.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg12/XBLOMShadowTreeEvent.java
```
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/AbstractSVGNumberList.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/AbstractSVGItem.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/AbstractSVGNormPathSegList.java
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/SVGAnimationContext.java

No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Qualified Name Module ......................................... -->
<!-- file: svg-qname.mod

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-qname.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ENTITIES SVG 1.1 Qualified Name//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-qname.mod"

....................................................................... -->

<!-- Qualified Name

This module is contained in two parts, labeled Section 'A' and 'B':

Section A declares parameter entities to support namespace-qualified names, namespace declarations, and name prefixing for SVG and extensions.

Section B declares parameter entities used to provide namespace-qualified names for all SVG element types:

-->

<!-- Section A: SVG XML Namespace Framework :..................... -->

<!-- 1. Declare a %SVG.prefixed; conditional section keyword, used to activate namespace prefixing. The default value should inherit '%NS.prefixed;' from the DTD driver, so that unless overridden, the default behaviour follows the overall DTD prefixing scheme.

-->

<!ENTITY % NS.prefixed "IGNORE" >
```
<!ENTITY % SVG.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %SVG.xmlns;) containing
       the URI reference used to identify the SVG namespace:
-->
<!ENTITY % SVG.xmlns "http://www.w3.org/2000/svg" >
<!ENTITY % XLINK.xmlns "http://www.w3.org/1999/xlink" >

<!-- 3. Declare parameter entities (eg., %SVG.prefix;) containing
       the default namespace prefix string(s) to use when prefixing
       is enabled. This may be overridden in the DTD driver or the
       internal subset of an document instance. If no default prefix
       is desired, this may be declared as an empty string.
-->
<!ENTITY % SVG.prefix "" >
<!ENTITY % XLINK.prefix "xlink" >

<!-- 4. Declare parameter entities (eg., %SVG.pfx;) containing the
       colonized prefix(es) (eg., '%SVG.prefix;:') used when
       prefixing is active, an empty string when it is not.
-->
<![%SVG.prefixed;[
 <ENTITY % SVG.pfx "%SVG.prefix;" >
 ]]>
<!ENTITY % SVG.pfx "" >
<!ENTITY % XLINK.pfx "%XLINK.prefix;" >

<!-- 5. The parameter entity %SVG.xmlns.extra.attrib; may be
       redeclared to contain any non-SVG namespace declaration
       attributes for namespaces embedded in SVG. The default
       is an empty string.
-->
<!ENTITY % SVG.xmlns.extra.attrib "" >

<!-- Declare a parameter entity XLINK.xmlns.attrib containing
       the XML Namespace declarations for XLink.
-->
<!ENTITY % XLINK.xmlns.attrib
 "xmlns:%XLINK.prefix; %URI.datatype; #FIXED '%XLINK.xmlns;'' >

<!-- Declare a parameter entity %NS.decl.attrib; containing
       all XML Namespace declarations used in the DTD, plus the
       xmlns declaration for SVG, its form dependent on whether
       prefixing is active.
-->
<![%SVG.prefixed;[
 <ENTITY % NS.decl.attrib
<!ENTITY % NS.decl.attrib "%XLINK.xmlns.attrib;
%SVG.xmlns.extra.attrib;">

<!ENTITY % SVG.xmlns.attrib "xmlns %URI.datatype; #FIXED '%SVG.xmlns;
%XLINK.xmlns.attrib;
%SVG.xmlns.extra.attrib;">

<!-- Declare a parameter entity %SVG.xmlns.attrib; containing all XML namespace declaration attributes used by SVG, including a default xmlns attribute when prefixing is inactive.
-->
<![%SVG.prefixed;[
<!ENTITY % SVG.xmlns.attrib "%NS.decl.attrib;">

<!ENTITY % SVG.svg.qname "%SVG.pfx;svg" >
<!ENTITY % SVG.g.qname "%SVG.pfx:g" >
<!ENTITY % SVG.defs.qname "%SVG.pfx:defs" >
<!ENTITY % SVG.desc.qname "%SVG.pfx:desc" >
<!ENTITY % SVG.title.qname "%SVG.pfx:title" >
<!ENTITY % SVG.metadata.qname "%SVG.pfx:metadata" >
<!ENTITY % SVG.symbol.qname "%SVG.pfx:symbol" >
<!ENTITY % SVG.use.qname "%SVG.pfx:use" >

<!-- module: svg-structure.mod ......................... -->

<!-- module: svg-conditional.mod ....................... -->

<!-- module: svg-image.mod ............................. -->
<!-- module: svg-animation.mod ......................... -->

<!ENTITY % SVG.animate.qname "%SVG.pfx;animate" >
<!ENTITY % SVG.set.qname "%SVG.pfx;set" >
<!ENTITY % SVG.animateMotion.qname "%SVG.pfx;animateMotion" >
<!ENTITY % SVG.animateColor.qname "%SVG.pfx;animateColor" >
<!ENTITY % SVG.animateTransform.qname "%SVG.pfx;animateTransform" >
<!ENTITY % SVG.mpath.qname "%SVG.pfx;mpath" >

<!-- module: svg-font.mod .............................. -->

<!ENTITY % SVG.font.qname "%SVG.pfx;font" >
<!ENTITY % SVG.font-face.qname "%SVG.pfx;font-face" >
<!ENTITY % SVG.glyph.qname "%SVG.pfx;glyph" >
<!ENTITY % SVG.missing-glyph.qname "%SVG.pfx;missing-glyph" >
<!ENTITY % SVG.hkern.qname "%SVG.pfx;hkern" >
<!ENTITY % SVG.vkern.qname "%SVG.pfx;vkern" >
<!ENTITY % SVG.font-face-src.qname "%SVG.pfx;font-face-src" >
<!ENTITY % SVG.font-face-uri.qname "%SVG.pfx;font-face-uri" >
<!ENTITY % SVG.font-face-format.qname "%SVG.pfx;font-face-format" >
<!ENTITY % SVG.font-face-name.qname "%SVG.pfx;font-face-name" >
<!ENTITY % SVG.definition-src.qname "%SVG.pfx;definition-src" >

<!-- module: svg-extensibility.mod ..................... -->

<!ENTITY % SVG.foreignObject.qname "%SVG.pfx;foreignObject" >

<!ENTITY % SVG.foreignObject.qname "%SVG.pfx;foreignObject" >

<!-- end of svg-qname.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg/resources/svg-qname.mod
No license file was found, but licenses were detected in source scan.

<!-- .................svg-opacity-attrib.mod ..................... -->

<!-- SVG 1.1 Paint Opacity Attribute Module ......................... -->
<!-- file: svg-opacity-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-opacity-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ENTITIES SVG 1.1 Paint Opacity Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-opacity-attrib.mod"
<!-- Paint Opacity Attribute

opacity, fill-opacity, stroke-opacity

This module defines the Opacity attribute set. -->

<!ENTITY % SVG.opacity.attrib
   "opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.fill-opacity.attrib
   "fill-opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-opacity.attrib
   "stroke-opacity %OpacityValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.Opacity.extra.attrib ""
>

<!ENTITY % SVG.Opacity.attrib
   "%SVG.opacity.attrib;
   %SVG.fill-opacity.attrib;
   %SVG.stroke-opacity.attrib;
   %SVG.Opacity.extra.attrib;"
>

<!-- end of svg-opacity-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg/resources/svg-opacity-attrib.mod
No license file was found, but licenses were detected in source scan.

<!-- .............................................................. -->

<!-- SVG 1.1 Basic Attribute Collection Module ......................... -->

<!-- file: svg11-basic-attribs.mod

This is SVG Basic, a proper subset of SVG.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-basic-attrs.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ENTITIES SVG 1.1 Basic Attribute Collection//EN"
<!-- SVG 1.1 Basic Attribute Collection -->

This module defines the set of common attributes that can be present on many SVG elements.

<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % ExtensionList.datatype "CDATA" >
<!ENTITY % FeatureList.datatype "CDATA" >

<!ENTITY % SVG.Conditional.extra.attrib "" >
<!ENTITY % SVG.Conditional.attrib
  "requiredFeatures %FeatureList.datatype; #IMPLIED
  requiredExtensions %ExtensionList.datatype; #IMPLIED
  systemLanguage %LanguageCodes.datatype; #IMPLIED
  %SVG.Conditional.extra.attrib;"
>

<!-- module: svg-style.mod ............................. -->

<!ENTITY % ClassList.datatype "CDATA" >
<!ENTITY % StyleSheet.datatype "CDATA" >

<!ENTITY % SVG.Style.extra.attrib "" >
<!ENTITY % SVG.Style.attrib
  "style %StyleSheet.datatype; #IMPLIED
  class %ClassList.datatype; #IMPLIED
  %SVG.Style.extra.attrib;"
>

<!-- module: svg-text.mod .............................. -->

<!ENTITY % BaselineShiftValue.datatype "CDATA" >
<!ENTITY % FontFamilyValue.datatype "CDATA" >
<!ENTITY % FontSizeValue.datatype "CDATA" >
<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >
<!ENTITY % KerningValue.datatype "CDATA" >
<!ENTITY % SpacingValue.datatype "CDATA" >
<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % SVG.Text.extra.attrib "" >
<!ENTITY % SVG.Text.attrib "writing-mode ( lr-tb | rl-tb | tb-rl | lr | rl | tb | inherit ) #IMPLIED
%SVG.Text.extra.attrib;">

<!ENTITY % SVG.TextContent.extra.attrib "" >
<!ENTITY % SVG.TextContent.attrib "alignment-baseline ( auto | baseline | before-edge | text-before-edge | middle | central | after-edge | text-after-edge | ideographic | alphabetic | hanging | mathematical | inherit ) #IMPLIED
baseline-shift %BaselineShiftValue.datatype; #IMPLIED
direction ( ltr | rtl | inherit ) #IMPLIED
dominant-baseline ( auto | use-script | no-change | reset-size | ideographic | alphabetic | hanging | mathematical | central | middle | text-after-edge | text-before-edge | inherit ) #IMPLIED
glyph-orientation-horizontal %GlyphOrientationHorizontalValue.datatype;
#IMPLIED
glyph-orientation-vertical %GlyphOrientationVerticalValue.datatype;
#IMPLIED
kerning %KerningValue.datatype; #IMPLIED
letter-spacing %SpacingValue.datatype; #IMPLIED
text-anchor ( start | middle | end | inherit ) #IMPLIED
text-decoration %TextDecorationValue.datatype; #IMPLIED
unicode-bidi ( normal | embed | bidi-override | inherit ) #IMPLIED
word-spacing %SpacingValue.datatype; #IMPLIED
%SVG.TextContent.extra.attrib;">

<!ENTITY % SVG.Font.extra.attrib "" >
<!ENTITY % SVG.Font.attrib "font-family %FontFamilyValue.datatype; #IMPLIED
font-size %FontSizeValue.datatype; #IMPLIED
font-size-adjust %FontSizeAdjustValue.datatype; #IMPLIED
font-stretch ( normal | wider | narrower | ultra-condensed |
extra-condensed | condensed | semi-condensed |
semi-expanded | expanded | extra-expanded |
ultra-expanded | inherit ) #IMPLIED
font-style ( normal | italic | oblique | inherit ) #IMPLIED
font-variant ( normal | small-caps | inherit ) #IMPLIED
font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 |
500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
%SVG.Font.extra.attrib;">

<!--[-- module: svg-profile.mod ................. -->}
<!ENTITY % SVG.Profile.extra.attrib "" >
<!ENTITY % SVG.Profile.attrib 
"color-profile CDATA #IMPLIED 
%SVG.Profile.extra.attrib;" >

<!-- module: svg-gradient.mod ......................... -->

<!ENTITY % NumberOrPercentage.datatype "CDATA" >

<!ENTITY % SVG.Gradient.extra.attrib "" >
<!ENTITY % SVG.Gradient.attrib 
"stop-color %SVGColor.datatype; #IMPLIED 
stop-opacity %OpacityValue.datatype; #IMPLIED 
%SVG.Gradient.extra.attrib;" >

<!-- module: svg-basic-clip.mod ....................... -->

<!ENTITY % ClipPathValue.datatype "CDATA" >

<!ENTITY % SVG.Clip.extra.attrib "" >
<!ENTITY % SVG.Clip.attrib 
"clip-path %ClipPathValue.datatype; #IMPLIED 
clip-rule %ClipFillRule.datatype; #IMPLIED 
%SVG.Clip.extra.attrib;" >

<!-- module: svg-mask.mod ............................. -->

<!ENTITY % MaskValue.datatype "CDATA" >

<!ENTITY % SVG.Mask.extra.attrib "" >
<!ENTITY % SVG.Mask.attrib 
"mask %MaskValue.datatype; #IMPLIED 
%SVG.Mask.extra.attrib;" >

<!-- module: svg-basic-filter.mod ..................... -->

<!ENTITY % FilterValue.datatype "CDATA" >
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!ENTITY % SVG.Filter.extra.attrib "" >
<!ENTITY % SVG.Filter.attrib 
"filter %FilterValue.datatype; #IMPLIED 
%SVG.Filter.extra.attrib;" >
<ENTITY % SVG.FilterColor.extra.attrib "" >
<ENTITY % SVG.FilterColor.attrib
  "color-interpolation-filters ( auto | sRGB | linearRGB | inherit )
  #IMPLIED
%SVG.FilterColor.extra.attrib;" >

<!ENTITY % CursorValue.datatype "CDATA" >
<ENTITY % SVG.Cursor.extra.attrib "" >
<ENTITY % SVG.Cursor.attrib
  "cursor %CursorValue.datatype; #IMPLIED
%SVG.Cursor.extra.attrib;" >

<!-- end of svg11-basic-attribs.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg/resources/svg11-basic-attribs.mod
No license file was found, but licenses were detected in source scan.

<!-- ................................................................. -->
<!-- SVG 1.1 Basic Clip Module ......................................... -->
<!-- file: svg-basic-clip.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-basic-clip.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

    PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Clip//EN"
    SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-clip.mod"

................................................................. -->

<!-- Basic Clip

    clipPath

    This module declares markup to provide support for clipping. -->

<!-- 'clip-path' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % ClipPathValue.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.ClipPath.qname "clipPath" >

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!ENTITY % SVG.Clip.class "" >

<!ENTITY % SVG.Clip.extra.class "" >

<!ENTITY % SVG.Clip.class >
"
%SVG.ClipPath.qname; %SVG.Clip.extra.class;"

<!ENTITY % SVG.Clip.attrib "" >

<!ENTITY % SVG.Clipattrib >
"clip-path %ClipPathValue.datatype; #IMPLIED
clip-rule %ClipFillRule.datatype; #IMPLIED
%SVG.Clip.extra.attrib;"

<!ENTITY % SVG.ClipPath.element "INCLUDE" >
<![%SVG.ClipPath.element;]>
<!ENTITY % SVG.ClipPath.content>
Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/resources/svg-basic-clip.mod
No license file was found, but licenses were detected in source scan.

This is SVG, a language for describing two-dimensional graphics in XML. 
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. 
Revision: $Id: svg-docevents-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $ 

This DTD module is identified by the PUBLIC and SYSTEM identifiers: 

PUBLIC "-//W3C//ENTITIES SVG 1.1 Document Events Attribute//EN" 
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-docevents-attrib.mod"
<!-- Document Events Attribute

unload, onabort, onerror, onresize, onscroll, onzoom

This module defines the DocumentEvents attribute set.
-->

<!ENTITY % SVG.onunload.attrib
  "onunload %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onabort.attrib
  "onabort %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onerror.attrib
  "onerror %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onresize.attrib
  "onresize %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onscroll.attrib
  "onscroll %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onzoom.attrib
  "onzoom %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.DocumentEvents.extra.attrib "" >

<!ENTITY % SVG.DocumentEvents.attrib
  "%SVG.onunload.attrib;
  %SVG.onabort.attrib;
  %SVG.onerror.attrib;
  %SVG.onresize.attrib;
  %SVG.onscroll.attrib;
  %SVG.onzoom.attrib;
  %SVG.DocumentEvents.extra.attrib;"
>

<!-- end of svg-docevents-attrib.mod -->

Open Source Used In DNAC Platform Arica  254
Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/resources/svg-docevents-attrib.mod
No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Datatypes Module .............................................. -->
<!-- file: svg-datatypes.mod
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-datatypes.mod 1733416 2016-03-03 07:07:13Z gadams$ -->

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Datatypes//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-datatypes.mod"

... ....................................................................... -->

<!-- Datatypes
This module declares common data types for properties and attributes. -->

<!-- feature specification -->
<!ENTITY % Boolean.datatype "( false | true )" >

<!-- 'clip-rule' or 'fill-rule' property/attribute value -->
<!ENTITY % ClipFillRule.datatype "( nonzero | evenodd | inherit )" >

<!-- media type, as per [RFC2045] -->
<!ENTITY % ContentType.datatype "CDATA" >

<!-- a <coordinate> -->
<!ENTITY % Coordinate.datatype "CDATA" >

<!-- a list of <coordinate>s -->
<!ENTITY % Coordinates.datatype "CDATA" >

<!-- a <color> value -->
<!ENTITY % Color.datatype "CDATA" >

<!-- a <integer> -->
<!ENTITY % Integer.datatype "CDATA" >

<!-- a language code, as per [RFC3066] -->
<!ENTITY % LanguageCode.datatype "NMTOKEN" >
No license file was found, but licenses were detected in source scan.

<!-- ................................................................. -->
<!-- SVG 1.1 Graphical Element Events Attribute Module ............... -->
<!-- file: svg-graphevents-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-graphevents-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "/W3C//ENTITIES SVG 1.1 Graphical Element Events Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-graphevents-attrib.mod"

................................................................. -->

<!-- Graphical Element Events Attribute

onfocusin, onfocusout, onactivate, onclick, onmousedown, onmouseup,
onmouseover, onmousemove, onmouseout, onload

This module defines the GraphicalEvents attribute set. -->

<!ENTITY % SVG.onfocusin.attrib
    "onfocusin %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onfocusout.attrib
    "onfocusout %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onactivate.attrib
    "onactivate %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onclick.attrib
    "onclick %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmousedown.attrib
    "onmousedown %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseup.attrib
    "onmouseup %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmousemove.attrib
    "onmousemove %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseout.attrib
    "onmouseout %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onmouseover.attrib
    "onmouseover %Script.datatype; #IMPLIED"
>

<!ENTITY % SVG.onload.attrib
    "onload %Script.datatype; #IMPLIED"
>
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-pattern.mod 1733416 2016-03-03 07:07:13Z gadams $
This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "//W3C//ELEMENTS SVG 1.1 Pattern//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-pattern.mod"

.............................................................. -->

<!-- Pattern

pattern

This module declares markup to provide support for pattern fill.

-->

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.pattern.qname "pattern" >

<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Pattern.class ......................... -->

<!ENTITY % SVG.Pattern.extra.class "" >

<!ENTITY % SVG.Pattern.class
Open Source Used In DNAC Platform Arica 261

<ENTITY % SVG.pattern.attlist "INCLUDE" >
<![%SVG.pattern.attlist;]
<!ATTLIST %SVG.pattern.qname;
  %SVG.Core.attrib;
  %SVG.Conditional.attrib;
  %SVG.Style.attrib;
  %SVG.Presentation.attrib;
  %SVG.XLink.attrib;
  %SVG.External.attrib;
  x %Coordinate.datatype; #IMPLIED
  y %Coordinate.datatype; #IMPLIED
  width %Length.datatype; #IMPLIED
  height %Length.datatype; #IMPLIED
  patternUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  patternContentUnits ( userSpaceOnUse | objectBoundingBox ) #IMPLIED
  patternTransform %TransformList.datatype; #IMPLIED
  viewBox %ViewBoxSpec.datatype; #IMPLIED
  preserveAspectRatio %PreserveAspectRatioSpec.datatype; 'xMidYMid meet'
>
<!-- end of SVG.pattern.attlist -->]]>
<!-- end of svg-pattern.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
  jar/org/apache/batik/dom/svg/resources/svg-pattern.mod
No license file was found, but licenses were detected in source scan.

<!!-- ................................................................. -->
<!!-- SVG 1.1 Attribute Collection Module ......................... -->
<!!-- file: svg11-attrs.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-attrs.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Attribute Collection//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-attrs.mod"

<!!-- SVG 1.1 Attribute Collection

This module defines the set of common attributes that can be present
on many SVG elements.
<!-- module: svg-conditional.mod ....................... -->

<!ENTITY % ExtensionList.datatype "CDATA" >
<!ENTITY % FeatureList.datatype "CDATA" >

<!ENTITY % SVG.Conditional.extra.attrib "" >
<!ENTITY % SVG.Conditional.attrib
"requiredFeatures %FeatureList.datatype; #IMPLIED
requiredExtensions %ExtensionList.datatype; #IMPLIED
systemLanguage %LanguageCodes.datatype; #IMPLIED
%SVG.Conditional.extra.attrib;" >

<!-- module: svg-style.mod ............................. -->

<!ENTITY % ClassList.datatype "CDATA" >
<!ENTITY % StyleSheet.datatype "CDATA" >

<!ENTITY % SVG.Style.extra.attrib "" >
<!ENTITY % SVG.Style.attrib
"style %StyleSheet.datatype; #IMPLIED
class %ClassList.datatype; #IMPLIED
%SVG.Style.extra.attrib;" >

<!-- module: svg-text.mod .............................. -->

<!ENTITY % BaselineShiftValue.datatype "CDATA" >
<!ENTITY % FontFamilyValue.datatype "CDATA" >
<!ENTITY % FontSizeValue.datatype "CDATA" >
<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >
<!ENTITY % KerningValue.datatype "CDATA" >
<!ENTITY % SpacingValue.datatype "CDATA" >
<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % SVG.Text.extra.attrib "" >
<!ENTITY % SVG.Text.attrib
"writing-mode ( lr-tb | rl-tb | tb-rl | lr | rl | tb | inherit ) #IMPLIED
%SVG.Text.extra.attrib;" >

<!ENTITY % SVG.TextContent.extra.attrib "" >
<!ENTITY % SVG.TextContent.attrib
"alignment-baseline ( auto | baseline | before-edge | text-before-edge |
middle | central | after-edge | text-after-edge |
ideographic | alphabetic | hanging | mathematical |
inherit ) #IMPLIED
baseline-shift %BaselineShiftValue.datatype; #IMPLIED
direction ( ltr | rtl | inherit ) #IMPLIED
dominant-baseline ( auto | use-script | no-change | reset-size |
ideographic | alphabetic | hanging | mathematical |
central | middle | text-after-edge | text-before-edge |
inherit ) #IMPLIED
glyph-orientation-horizontal %GlyphOrientationHorizontalValue.datatype; 
#IMPLIED
glyph-orientation-vertical %GlyphOrientationVerticalValue.datatype; 
#IMPLIED
kerning %KerningValue.datatype; #IMPLIED
letter-spacing %SpacingValue.datatype; #IMPLIED
text-anchor ( start | middle | end | inherit ) #IMPLIED
text-decoration %TextDecorationValue.datatype; #IMPLIED
unicode-bidi ( normal | embed | bidi-override | inherit ) #IMPLIED
word-spacing %SpacingValue.datatype; #IMPLIED
%SVG.TextContent.extra.attrib;"
>
<!ENTITY % SVG.Font.extra.attrib "" >
<!ENTITY % SVG.Font.attrib
"font-family %FontFamilyValue.datatype; #IMPLIED
font-size %FontSizeValue.datatype; #IMPLIED
font-size-adjust %FontSizeAdjustValue.datatype; #IMPLIED
font-stretch ( normal | wider | narrower | ultra-condensed |
exasy-condensed | condensed | semi-condensed |
semi-expanded | expanded | extra-expanded |
ultra-expanded | inherit ) #IMPLIED
font-style ( normal | italic | oblique | inherit ) #IMPLIED
font-variant ( normal | small-caps | inherit ) #IMPLIED
font-weight ( normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | inherit ) #IMPLIED
%SVG.Font.extra.attrib;"
>
<!-- module: svg-marker.mod ......................... -->
<!ENTITY % MarkerValue.datatype "CDATA" >
<!ENTITY % SVG.Marker.extra.attrib "" >
<!ENTITY % SVG.Marker.attrib
"marker-start %MarkerValue.datatype; #IMPLIED
marker-mid %MarkerValue.datatype; #IMPLIED
marker-end %MarkerValue.datatype; #IMPLIED
%SVG.Marker.extra.attrib;"
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-filter.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C/ELEMENTS SVG 1.1 Filter//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-filter.mod"

filter, feBlend, feColorMatrix, feComponentTransfer, feComposite, feConvolveMatrix, feDiffuseLighting, feDisplacementMap, feFlood,
feGaussianBlur, feImage, feMerge, feMergeNode, feMorphology, feOffset,
feSpecularLighting, feTile, feTurbulence, feDistantLight, fePointLight,
feSpotLight, feFuncR, feFuncG, feFuncB, feFuncA

This module declares markup to provide support for filter effect.

-->

<!-- 'filter' property/attribute value (e.g., 'none', <uri>) -->
<!ENTITY % FilterValue.datatype "CDATA" >

<!-- list of <number>s, but at least one and at most two -->
<!ENTITY % NumberOptionalNumber.datatype "CDATA" >

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.filter.qname "filter" >
<!ENTITY % SVG.feBlend.qname "feBlend" >
<!ENTITY % SVG.feColorMatrix.qname "feColorMatrix" >
<!ENTITY % SVG.feComponentTransfer.qname "feComponentTransfer" >
<!ENTITY % SVG.feComposite.qname "feComposite" >
<!ENTITY % SVG.feConvolveMatrix.qname "feConvolveMatrix" >
<!ENTITY % SVG.feDiffuseLighting.qname "feDiffuseLighting" >
<!ENTITY % SVG.feDisplacementMap.qname "feDisplacementMap" >
<!ENTITY % SVG.feFlood.qname "feFlood" >
<!ENTITY % SVG.feGaussianBlur.qname "feGaussianBlur" >
<!ENTITY % SVG.feImage.qname "feImage" >
<!ENTITY % SVG.feMerge.qname "feMerge" >
<!ENTITY % SVG.feMergeNode.qname "feMergeNode" >
<!ENTITY % SVG.feMorphology.qname "feMorphology" >
<!ENTITY % SVG.feOffset.qname "feOffset" >
<!ENTITY % SVG.feSpecularLighting.qname "feSpecularLighting" >
<!ENTITY % SVG.feTile.qname "feTile" >
<!ENTITY % SVG.feTurbulence.qname "feTurbulence" >
<!ENTITY % SVG.feDistantLight.qname "feDistantLight" >
<!ENTITY % SVG.fePointLight.qname "fePointLight" >
<!ENTITY % SVG.feSpotLight.qname "feSpotLight" >
<!ENTITY % SVG.feFuncR.qname "feFuncR" >
<!ENTITY % SVG.feFuncG.qname "feFuncG" >
<!ENTITY % SVG.feFuncB.qname "feFuncB" >
<!ENTITY % SVG.feFuncA.qname "feFuncA" >

<!-- Attribute Collections (Default) ..................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
x %Number.datatype; #IMPLIED
y %Number.datatype; #IMPLIED
z %Number.datatype; #IMPLIED
pointsAtX %Number.datatype; #IMPLIED
pointsAtY %Number.datatype; #IMPLIED
pointsAtZ %Number.datatype; #IMPLIED
specularExponent %Number.datatype; #IMPLIED
limitingConeAngle %Number.datatype; #IMPLIED
>}

<!-- end of SVG.feSpotLight.attlist --></!!>

<!-- feFuncR: Filter Effect Function Red Element ....... -->

<!ENTITY % SVG.feFuncR.extra.content "" >

<!ENTITY % SVG.feFuncR.element "INCLUDE" >
<![%SVG.feFuncR.element;]
<!ENTITY % SVG.feFuncR.content
  "(%SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncR.extra.content; )""
>
<!ELEMENT %SVG.feFuncR.qname; %SVG.feFuncR.content; >
<!-- end of SVG.feFuncR.element --></!!>

<!ENTITY % SVG.feFuncR.attlist "INCLUDE" >
<![%SVG.feFuncR.attlist;]
<!ATTLIST %SVG.feFuncR.qname;
  %SVG.Core.attrib;
    type ( identity | table | discrete | linear | gamma ) #REQUIRED
tableValues CDATA #IMPLIED
  slope %Number.datatype; #IMPLIED
  intercept %Number.datatype; #IMPLIED
  amplitude %Number.datatype; #IMPLIED
  exponent %Number.datatype; #IMPLIED
  offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncR.attlist --></!!>

<!-- feFuncG: Filter Effect Function Green Element .... -->

<!ENTITY % SVG.feFuncG.extra.content "" >

<!ENTITY % SVG.feFuncG.element "INCLUDE" >
<![%SVG.feFuncG.element;]
<!ENTITY % SVG.feFuncG.content
  "(%SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncG.extra.content; )""
>
<!ELEMENT %SVG.feFuncG.qname; %SVG.feFuncG.content; >
<!-- end of SVG.feFuncG.element --></!!>
Open Source Used In DNAC Platform Arica 281
"(%SVG.animate.qname; | %SVG.set.qname; %SVG.feFuncA.extra.content; )*"

<!ELEMENT %SVG.feFuncA.qname; %SVG.feFuncA.content; >
<!-- end of SVG.feFuncA.element -->]]>

<!ENTITY % SVG.feFuncA.attlist "INCLUDE" >
<![%SVG.feFuncA.attlist;]
<!ATTLIST %SVG.feFuncA.qname;
 %SVG.Core.attrib;
type ( identity | table | discrete | linear | gamma ) #REQUIRED
tableValues CDATA #IMPLIED
slope %Number.datatype; #IMPLIED
intercept %Number.datatype; #IMPLIED
amplitude %Number.datatype; #IMPLIED
exponent %Number.datatype; #IMPLIED
offset %Number.datatype; #IMPLIED
>
<!-- end of SVG.feFuncA.attlist -->]]>

<!-- end of svg-filter.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-jar/org/apache/batik/dom/svg/resources/svg-filter.mod
No license file was found, but licenses were detected in source scan.

dimensional graphics in XML.

The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg12

dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg12
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-basic-filter.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C/ELEMENTS SVG 1.1 Basic Filter//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-filter.mod"

....................................................................... -->

<!-- Basic Filter

filter, feBlend, feColorMatrix, feComponentTransfer, feComposite,
feFlood, feGaussianBlur, feImage, feMerge, feMergeNode, feOffset,
feTile, feFuncR, feFuncG, feFuncB, feFuncA

This module declares markup to provide support for filter effect. -->

<!ENTITY % FilterValue.datatype "CDATA" >

<ENTITY % NumberOptionalNumber.datatype "CDATA" >

<ENTITY % SVG.filter.qname "filter" >
<ENTITY % SVG.feBlend.qname "feBlend" >
<ENTITY % SVG.feColorMatrix.qname "feColorMatrix" >
<ENTITY % SVG.feComponentTransfer.qname "feComponentTransfer" >
<ENTITY % SVG.feComposite.qname "feComposite" >
<ENTITY % SVG.feFlood.qname "feFlood" >
<ENTITY % SVG.feGaussianBlur.qname "feGaussianBlur" >
<ENTITY % SVG.feImage.qname "feImage" >
<ENTITY % SVG.feMerge.qname "feMerge" >
<ENTITY % SVG.feMergeNode.qname "feMergeNode" >
<ENTITY % SVG.feOffset.qname "feOffset" >
<!ENTITY % SVG.feTile.qname "feTile" >
<!ENTITY % SVG.feFuncR.qname "feFuncR" >
<!ENTITY % SVG.feFuncG.qname "feFuncG" >
<!ENTITY % SVG.feFuncB.qname "feFuncB" >
<!ENTITY % SVG.feFuncA.qname "feFuncA" >

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!ENTITY % SVG.Filter.class ......................... -->

<!ENTITY % SVG.Filter.extra.class "" >

<!ENTITY % SVG.Filter.class

"| %SVG.filter.qname; %SVG.Filter.extra.class;"
>

<!-- SVG.FilterPrimitive.class ....................... -->

<!ENTITY % SVG.FilterPrimitive.class

"| %SVG.feBlend.qname; | %SVG.feColorMatrix.qname;
| %SVG.feComponentTransfer.qname; | %SVG.feComposite.qname;
| %SVG.feFlood.qname; | %SVG.feGaussianBlur.qname; | %SVG.feImage.qname;
| %SVG.feGaussianBlur.qname; | %SVG.feOffset.qname; | %SVG.feTile.qname;
| %SVG.FilterPrimitive.extra.class;"
"%SVG.Container.attrib; %SVG.Viewport.attrib; %SVG.Text.attrib; %SVG.TextContent.attrib; %SVG.Font.attrib; %SVG.Paint.attrib; %SVG.Color.attrib; %SVG.Opacity.attrib; %SVG.Graphics.attrib; %SVG.Marker.attrib; %SVG.Profile.attrib; %SVG.Gradient.attrib; %SVG.Clip.attrib; %SVG.Mask.attrib; %SVG.Filter.attrib; %SVG.FilterColor.attrib; %SVG.Cursor.attrib; flood-color %SVGColor.datatype; #IMPLIED flood-opacity %OpacityValue.datatype; #IMPLIED lighting-color %SVGColor.datatype; #IMPLIED %SVG.Presentation.extra.attrib;" >

<!-- filter: Filter Element ................................ -->

<!ENTITY % SVG.filter.extra.content "" >

<!ENTITY % SVG.filter.element "INCLUDE" >
<![%SVG.filter.element:]
<!ENTITY % SVG.filter.content "(( %SVG.Description.class; )*%SVG.animate.qname; | %SVG.set.qname; %SVG.FilterPrimitive.class; %SVG.filter.extra.content; )*)"
>

<!ELEMENT %SVG.filter.qname; %SVG.filter.content; >
<!-- end of SVG.filter.element -->]]>

<!ENTITY % SVG.filter.attlist "INCLUDE" >
<![%SVG.filter.attlist:][
<!ATTLIST %SVG.filter.qname; %SVG.Core.attrib; %SVG.Style.attrib; %SVG.Presentation.attrib; %SVG.XLink.attrib; %SVG.External.attrib; x %Coordinate.datatype; #IMPLIED y %Coordinate.datatype; #IMPLIED width %Length.datatype; #IMPLIED height %Length.datatype; #IMPLIED
type ( matrix | saturate | hueRotate | luminanceToAlpha ) 'matrix'
values CDATA #IMPLIED

<!-  end of SVG.feColorMatrix.attlist -->]]>

<!-  feComponentTransfer: Filter Effect Component Transfer Element -->

<!ENTITY % SVG.feComponentTransfer.extra.content "" >

<!ENTITY % SVG.feComponentTransfer.element "INCLUDE" >
<![%SVG.feComponentTransfer.element:]>
<!ENTITY % SVG.feComponentTransfer.content
"( %SVG.feFuncR.qname;?, %SVG.feFuncG.qname;?, %SVG.feFuncB.qname;?,
 %SVG.feFuncA.qname;? %SVG.feComponentTransfer.extra.content; )" >
<!ELEMENT %SVG.feComponentTransfer.qname; %SVG.feComponentTransfer.content; >
<!-  end of SVG.feComponentTransfer.element -->]

<!ENTITY % SVG.feComponentTransfer.attlist "INCLUDE" >
<![%SVG.feComponentTransfer.attlist:]>
<!ATTLIST %SVG.feComponentTransfer.qname;
 %SVG.Core.attrib;
 %SVG.FilterColor.attrib;
 %SVG.FilterPrimitiveWithIn.attrib;
 >
<!-  end of SVG.feComponentTransfer.attlist -->]]>

<!-  feComposite: Filter Effect Composite Element ...... -->

<!ENTITY % SVG.feComposite.extra.content "" >

<!ENTITY % SVG.feComposite.element "INCLUDE" >
<![%SVG.feComposite.element:]>
<!ENTITY % SVG.feComposite.content
"( %SVG.animate.qname; | %SVG.set.qname; %SVG.feComposite.extra.content; )*" >
<!ELEMENT %SVG.feComposite.qname; %SVG.feComposite.content; >
<!-  end of SVG.feComposite.element -->]]>

<!ENTITY % SVG.feComposite.attlist "INCLUDE" >
<![%SVG.feComposite.attlist:]>
<!ATTLIST %SVG.feComposite.qname;
 %SVG.Core.attrib;
 %SVG.FilterColor.attrib;
 %SVG.FilterPrimitiveWithIn.attrib;
in2 CDATA #REQUIRED
operator ( over | in | out | atop | xor | arithmetic ) 'over'
k1 %Number.datatype; #IMPLIED
The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
This paragraph appears in all copies. The copyright holders make no representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11.dtd 1733416 2016

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/resources/svg11.dtd

No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Paint Attribute Module ........................................ -->
<!-- file: svg-paint-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-paint-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ENTITIES SVG 1.1 Paint Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-paint-attrib.mod"

....................................................................... -->

<!-- Paint Attribute

fill, fill-rule, stroke, stroke-dasharray, stroke-dashoffset, stroke-linecap, stroke-linejoin, stroke-miterlimit, stroke-width, color, color-interpolation, color-rendering

This module defines the Paint and Color attribute sets.
-->

<!-- a 'fill' or 'stroke' property/attribute value: <paint> -->
<!ENTITY % Paint.datatype "CDATA" >

<!-- 'stroke-dasharray' property/attribute value (e.g., 'none', list of <number>s) -->
<!ENTITY % StrokeDashArrayValue.datatype "CDATA" >

<!-- 'stroke-dashoffset' property/attribute value (e.g., 'none', <length>) -->
<!ENTITY % StrokeDashOffsetValue.datatype "CDATA" >

<!-- 'stroke-miterlimit' property/attribute value (e.g., <number>) -->
<!ENTITY % StrokeMiterLimitValue.datatype "CDATA" >
<!- 'stroke-width' property/attribute value (e.g., <length>) -->
<!ENTITY % StrokeWidthValue.datatype "CDATA">

<!ENTITY % SVG.fill.attrib
  "fill %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.fill-rule.attrib
  "fill-rule %ClipFillRule.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke.attrib
  "stroke %Paint.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dasharray.attrib
  "stroke-dasharray %StrokeDashArrayValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-dashoffset.attrib
  "stroke-dashoffset %StrokeDashOffsetValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-linecap.attrib
  "stroke-linecap ( butt | round | square | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-linejoin.attrib
  "stroke-linejoin ( miter | round | bevel | inherit ) #IMPLIED"
>

<!ENTITY % SVG.stroke-miterlimit.attrib
  "stroke-miterlimit %StrokeMiterLimitValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.stroke-width.attrib
  "stroke-width %StrokeWidthValue.datatype; #IMPLIED"
>

<!ENTITY % SVG.Paint.extra.attrib ""
>

<!ENTITY % SVG.Paint.attrib
  "%SVG.fill.attrib;
  %SVG.fill-rule.attrib;
  %SVG.stroke.attrib;
  %SVG.stroke-dasharray.attrib;
  %SVG.stroke-dashoffset.attrib;
Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1- jar/org/apache/batik/dom/svg/resources/svg-paint-attrib.mod

No license file was found, but licenses were detected in source scan.

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-extresources-attrib.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ENTITIES SVG 1.1 External Resources Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-extresources-attrib.mod"
<!-- External Resources Attribute

externalResourcesRequired

This module defines the External attribute set.
-->

<!ENTITY % SVG.externalResourcesRequired.attrib "externalResourcesRequired %Boolean.datatype; #IMPLIED”>

<!ENTITY % SVG.External.extra.attrib "" >

<!ENTITY % SVG.External.attrib "%SVG.externalResourcesRequired.attrib;
%SVG.External.extra.attrib;”>

<!-- end of svg-extresources-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg-extresources-attrib.mod

No license file was found, but licenses were detected in source scan.

SVG 1.1 Basic DTD

This is SVG Basic, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-text.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Text//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-text.mod"

----------------------------------------

<!ENTITY % BaselineShiftValue.datatype "CDATA" >

<!ENTITY % FontFamilyValue.datatype "CDATA" >

<!ENTITY % FontSizeValue.datatype "CDATA" >

<!ENTITY % FontSizeAdjustValue.datatype "CDATA" >

<!ENTITY % GlyphOrientationHorizontalValue.datatype "CDATA" >

<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >
<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >

<!ENTITY % KerningValue.datatype "CDATA" >

<!ENTITY % SpacingValue.datatype "CDATA" >

<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % GlyphOrientationVerticalValue.datatype "CDATA" >

<!ENTITY % KerningValue.datatype "CDATA" >

<!ENTITY % SpacingValue.datatype "CDATA" >

<!ENTITY % TextDecorationValue.datatype "CDATA" >

<!ENTITY % SVG.text.qname "text" >
<!ENTITY % SVG.tspan.qname "tspan" >
<!ENTITY % SVG.tref.qname "tref" >
<!ENTITY % SVG.textPath.qname "textPath" >
<!ENTITY % SVG.altGlyph.qname "altGlyph" >
<!ENTITY % SVG.altGlyphDef.qname "altGlyphDef" >
<!ENTITY % SVG.altGlyphItem.qname "altGlyphItem" >
<!ENTITY % SVG.glyphRef.qname "glyphRef" >

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Conditional.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!ENTITY % SVG.Text.class "| %SVG.text.qname; | %SVG.altGlyphDef.qname; %SVG.Text.extra.class;" >
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-style.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ELEMENTS SVG 1.1 Style//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-style.mod"

This module declares markup to provide support for stylesheet.

<!-- Style

style

This module declares markup to provide support for stylesheet. -->

<!-- list of classes -->
<!ENTITY % ClassList.datatype "CDATA" >

<!-- comma-separated list of media descriptors. -->
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-basic-font.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Basic Font//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-font.mod"

................................. -->

<!-- Basic Font

font, font-face, glyph, missing-glyph, hkern, vkern, font-face-src,
font-face-uri, font-face-name

This module declares markup to provide support for template. -->

</!

<!ENTITY % SVG.font.qname "font" >
<!ENTITY % SVG.font-face.qname "font-face" >
<!ENTITY % SVG.glyph.qname "glyph" >
<!ENTITY % SVG.missing-glyph.qname "missing-glyph" >
<!ENTITY % SVG.hkern.qname "hkern" >
<!ENTITY % SVG.vkern.qname "vkern" >
<!ENTITY % SVG.font-face-src.qname "font-face-src" >
<!ENTITY % SVG.font-face-uri.qname "font-face-uri" >
<!ENTITY % SVG.font-face-name.qname "font-face-name" >
<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Container.attrib "" >
<!ENTITY % SVG.Style.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Text.attrib "" >
<!ENTITY % SVG.TextContent.attrib "" >
<!ENTITY % SVG.Font.attrib "" >
<!ENTITY % SVG.Paint.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Opacity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Marker.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Gradient.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.FilterColor.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkRequired.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Font.class ..................................... -->

<!ENTITY % SVG.Font.extra.class "" >

<!ENTITY % SVG.Font.class
"| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- SVG.Presentation.attrib .......................... -->

<!ENTITY % SVG.Presentation.extra.attrib "" >

<!ENTITY % SVG.Presentation.attrib
"%SVG.Container.attrib;
%SVG.Viewport.attrib;
%SVG.Text.attrib;
%SVG.TextContent.attrib;
%SVG.Font.attrib;
%SVG.Paint.attrib;
%SVG.Color.attrib;
%SVG.Opacity.attrib;
%SVG.Graphics.attrib;
%SVG.Marker.attrib;
%SVG.Profile.attrib;"
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-conditional.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C/ELEMENTS SVG 1.1 Conditional Processing//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-conditional.mod"

................................................................. -->

<!-- Conditional Processing -->

switch

This module declares markup to provide support for conditional processing.

-->
SVG 1.1 Basic DTD

This is SVG Basic, a proper subset of SVG.

The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg-conditional.mod
No license file was found, but licenses were detected in source scan.

<!-- ....................................................................... -->
<!-- SVG 1.1 Graphics Attribute Module ..................................... -->
<!-- file: svg-graphics-attrib.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-graphics-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Graphics Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-graphics-attrib.mod"

<!-- Graphics Attribute
display, image-rendering, pointer-events, shape-rendering, text-rendering, visibility

This module defines the Graphics attribute set.

-->

<!ENTITY % SVG.display.attrib "display ( inline | block | list-item | run-in | compact | marker |
    table | inline-table | table-row-group | table-header-group |
    table-footer-group | table-row | table-column-group |
    table-column | table-cell | table-caption | none | inherit ) #IMPLIED" >

<!ENTITY % SVG.image-rendering.attrib "image-rendering ( auto | optimizeSpeed | optimizeQuality | inherit ) #IMPLIED" >

<!ENTITY % SVG.pointer-events.attrib "pointer-events ( visiblePainted | visibleFill | visibleStroke | visible |
    painted | fill | stroke | all | none | inherit ) #IMPLIED" >

<!ENTITY % SVG.shape-rendering.attrib "shape-rendering ( auto | optimizeSpeed | crispEdges | geometricPrecision |
    inherit ) #IMPLIED" >

<!ENTITY % SVG.text-rendering.attrib "text-rendering ( auto | optimizeSpeed | optimizeLegibility |
    geometricPrecision | inherit ) #IMPLIED" >

<!ENTITY % SVG.visibility.attrib "visibility ( visible | hidden | inherit ) #IMPLIED" >

<!ENTITY % SVG.Graphics.extra.attrib "" >

<!ENTITY % SVG.Graphics.attrib "%SVG.display.attrib;
    %SVG.image-rendering.attrib;
    %SVG.pointer-events.attrib;
    %SVG.shape-rendering.attrib;
    %SVG.text-rendering.attrib;" >
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-xlink-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 XLink Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-xlink-attrib.mod"

.......................................................... -->

<!-- XLink Attribute

   type, href, role, arcrole, title, show, actuate

   This module defines the XLink, XLinkRequired, XLinkEmbed, and
   XLinkReplace attribute set. -->

<!ENTITY % SVG.XLink.extra.attrib "" >

<!ENTITY % SVG.XLink.attrib "%XLINK.xmlns.attrib;
   %XLINK.pfx:type ( simple ) #FIXED 'simple'
   %XLINK.pfx:href %URI.datatype; #IMPLIED
   %XLINK.pfx:role %URI.datatype; #IMPLIED
   %XLINK.pfx:arcrole %URI.datatype; #IMPLIED
   %XLINK.pfx:title CDATA #IMPLIED
   %XLINK.pfx:show ( other ) 'other'
   %XLINK.pfx:actuate ( onLoad ) #FIXED 'onLoad'
 %SVG.XLink.extra.attrib;" >
No license file was found, but licenses were detected in source scan.

<!-- .............................................................. -->
<!-- SVG 1.1 Scripting Module ...................................... -->
<!-- file: svg-script.mod

This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-script.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C/ELEMENTS SVG 1.1 Scripting//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-script.mod"

.............................................................. -->

<!-- Scripting

script

This module declares markup to provide support for scripting. -->

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.script.qname "script" >

<!-- Attribute Collections (Default) ................... -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.XLink.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Script.class .................................. 
<!ENTITY % SVG.Script.extra.class "" >

<!ENTITY % SVG.Script.class

"| %SVG.script.qname; %SVG.Script.extra.class;"
>

<!-- script: Script Element ............................ -->

<!ENTITY % SVG.script.extra.content "" >

<!ENTITY % SVG.script.element "INCLUDE" >
<![%SVG.script.element[}
This is SVG, a language for describing two-dimensional graphics in XML. Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved. Revision: $Id: svg-font.mod 1733416 2016-03-03 07:07:13Z gadams$

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ELEMENTS SVG 1.1 Font//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-font.mod"

This module declares markup to provide support for template.

....... Font..............

text, font-face, glyph, missing-glyph, hkern, vkern, font-face-src,
font-face-uri, font-face-format, font-face-name, definition-src

This module declares markup to provide support for template.

... Quali... (Default) ...............
Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1.jar/org/apache/batik/dom/svg/resources/svg-font.mod
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

SVG 1.1 Tiny DTD

This is SVG Tiny, a proper subset of SVG.
The Scalable Vector Graphics (SVG)
(Massachusetts Institute of Technology, Institut National de
Recherche en Informatique et en Automatique, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the SVG DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Jun Fujisawa <fujisawa.jun@canon.co.jp>
Revision: $Id: svg11

 dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg11-tiny-flat.dtd
No license file was found, but licenses were detected in source scan.

<!- .............................................................. -->
<!- SVG 1.1 Tiny Document Model Module ......................... -->
<!- file: svg11-tiny-model.mod

This is SVG Tiny, a proper subset of SVG.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg11-tiny-model.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES SVG 1.1 Tiny Document Model//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11-tiny-model.mod"

.............................................................. -->

<!- SVG 1.1 Tiny Document Model

This module describes the groupings of elements that make up
common content models for SVG elements.
-->

<!- Content Models (Default) ......................... -->
"%SVG.animate.qname; | %SVG.set.qname; | %SVG.animateMotion.qname; |
%SVG.animateColor.qname; | %SVG.animateTransform.qname;
%SVG.Animation.extra.class;";

<!-- module: svg-basic-font.mod .................. -->

<!ENTITY % SVG.Font.extra.class "" >
<!ENTITY % SVG.Font.class
"| %SVG.font.qname; | %SVG.font-face.qname; %SVG.Font.extra.class;"
>

<!-- module: svg-extensibility.mod .............. -->

<!ENTITY % SVG.Extensibility.extra.class "" >
<!ENTITY % SVG.Extensibility.class
"| %SVG.foreignObject.qname; %SVG.Extensibility.extra.class;"
>

<!-- end of svg11-tiny-model.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg11-tiny-model.mod
No license file was found, but licenses were detected in source scan.

<!-- .......................................................... -->
<!-- SVG 1.1 Animation Module ............................. -->
<!-- file: svg-animation.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-animation.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ELEMENTS SVG 1.1 Animation//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-animation.mod"

.......................................................... -->

<!-- Animation

animate, set, animateMotion, animateColor, animateTransform, mpath

This module declares markup to provide support for animation.
-->
<!ENTITY % SVG.AnimationTiming.extra.attrib "" >

<!ENTITY % SVG.AnimationTiming.attrib 
"begin CDATA #IMPLIED
  dur CDATA #IMPLIED
  end CDATA #IMPLIED
  min CDATA #IMPLIED
  max CDATA #IMPLIED
  restart ( always | never | whenNotActive ) 'always'
  repeatCount CDATA #IMPLIED
  repeatDur CDATA #IMPLIED
  fill ( remove | freeze ) 'remove'
%SVG.AnimationTiming.extra.attrib;" >

<!ENTITY % SVG.AnimationValue.attrib ......................... 
"calcMode ( discrete | linear | paced | spline ) 'linear'
  values CDATA #IMPLIED
  keyTimes CDATA #IMPLIED
  keySplines CDATA #IMPLIED
  from CDATA #IMPLIED
  to CDATA #IMPLIED
  by CDATA #IMPLIED
%SVG.AnimationValue.extra.attrib;" >

<!ENTITY % SVG.AnimationAddtion.extra.attrib "" >

<!ENTITY % SVG.AnimationAddtion.attrib 
"additive ( replace | sum ) 'replace'
  accumulate ( none | sum ) 'none'
%SVG.AnimationAddtion.extra.attrib;" >

<! ENTITY animate: Animate Element ......................... -->

<!ENTITY % SVG.animate.extra.content "" >

<!ENTITY % SVG.animate.element "INCLUDE" >
<![%SVG.animate.element;]<!ENTITY % SVG.animate.content
This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-viewport-attrib.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "//W3C//ENTITIES SVG 1.1 Viewport Attribute//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-viewport-attrib.mod"

----------------------------------------------- --

Viewport Attribute

clip, overflow

This module defines the Viewport attribute set.
-->

<!-- 'clip' property/attribute value (e.g., 'auto', rect(...)) -->
<!ENTITY % ClipValue.datatype "CDATA" >

<!ENTITY % SVG.clip.attrib
"clip %ClipValue.datatype; #IMPLIED"
>
<!ENTITY % SVG.overflow.attrib
"overflow ( visible | hidden | scroll | auto | inherit ) #IMPLIED"
>
<!ENTITY % SVG.Viewport.extra.attrib "" >

<!ENTITY % SVG.Viewport.attrib
"%SVG.clip.attrib;
%SVG.overflow.attrib;
%SVG.Viewport.extra.attrib;"
>
</!-- end of svg-viewport-attrib.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1-
jar/org/apache/batik/dom/svg/resources/svg-viewport-attrib.mod
No license file was found, but licenses were detected in source scan.

</!-- ............................................................... -->
</!-- SVG 1.1 Basic Text Module ..................................... -->
</!-- file: svg-basic-text.mod

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-basic-text.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C//ELEMENTS SVG 1.1 Basic Text//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-basic-text.mod"

............................................................... -->

</!-- Basic Text

text, altGlyph, altGlyphDef, glyphRef

This module declares markup to provide support for text.
-->
Open Source Used In DNAC Platform Arica 348

> "Open Source Used In DNAC Platform Arica 348"

< !-- end of SVG.altGlyphDef.attlist --> ]]>

<!- glyphRef: Glyph Reference Element ............... -->

<!ENTITY % SVG.glyphRef.element "INCLUDE" >
<![%SVG.glyphRef.element;
<!ENTITY % SVG.glyphRef.content "EMPTY" >
<!ELEMENT %SVG.glyphRef.qname; %SVG.glyphRef.content; >
<!- end of SVG.glyphRef.element --> ]]>

<!ENTITY % SVG.glyphRef.attlist "INCLUDE" >
<![%SVG.glyphRef.attlist;
<!ATTLIST %SVG.glyphRef.qname;
 %SVG.Core.attrib;
 %SVG.Style.attrib;
 %SVG.Font.attrib;
 %SVG.XLink.attrib;
 glyphRef CDATA #IMPLIED
 format CDATA #IMPLIED
 >
<!- end of SVG.glyphRef.attlist --> ]]>

<!- end of svg-basic-text.mod -->

Found in path(s):
* /opt/cola/permits/1146878524_1616835142.61/0/batik-svg-dom-1-13-sources-1- jar/org/apache/batik/dom/svg/resources/svg-basic-text.mod
No license file was found, but licenses were detected in source scan.

<!- .......................................................... -->
<!- SVG 1.1 Image Module ..................................... -->
<!- file: svg-image.mod -->

This is SVG, a language for describing two-dimensional graphics in XML.
Copyright 2001, 2002 W3C (MIT, INRIA, Keio), All Rights Reserved.
Revision: $Id: svg-image.mod 1733416 2016-03-03 07:07:13Z gadams $

This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-/W3C/ELEMENTS SVG 1.1 Image//EN"
SYSTEM "http://www.w3.org/Graphics/SVG/1.1/DTD/svg-image.mod"

.......................................................... -->

<!- Image -->

image
This module declares markup to provide support for image.

<!-- Qualified Names (Default) ......................... -->

<!ENTITY % SVG.image.qname "image" >

<!-- Attribute Collections (Default) ................. -->

<!ENTITY % SVG.Core.attrib "" >
<!ENTITY % SVG.Condition.attrib "" >
<!ENTITY % SVG.Viewport.attrib "" >
<!ENTITY % SVG.Color.attrib "" >
<!ENTITY % SVG.Popularity.attrib "" >
<!ENTITY % SVG.Graphics.attrib "" >
<!ENTITY % SVG.Profile.attrib "" >
<!ENTITY % SVG.Clip.attrib "" >
<!ENTITY % SVG.Mask.attrib "" >
<!ENTITY % SVG.Filter.attrib "" >
<!ENTITY % SVG.GraphicalEvents.attrib "" >
<!ENTITY % SVG.Cursor.attrib "" >
<!ENTITY % SVG.XLinkEmbed.attrib "" >
<!ENTITY % SVG.External.attrib "" >

<!-- SVG.Image.class ................................... -->

<!ENTITY % SVG.Image.extra.class "" >
<!ENTITY % SVG.Image.class "| %SVG.image.qname; %SVG.Image.extra.class;" >

<!-- image: Image Element .............................. -->

<!ENTITY % SVG.image.extra.content "" >

<!ENTITY % SVG.image.element "INCLUDE" >
><![%SVG.image.element;[
<!ENTITY % SVG.image.content "(( %SVG.Description.class; )*, ( %SVG.Animation.class; %SVG.image.extra.content; ))" >
<!ELEMENT %SVG.image.qname; %SVG.image.content; >
<!!-- end of SVG.image.element -->]]>

<!ENTITY % SVG.image.attlist "INCLUDE" >
1.16 metrics-jetty 4.1.0

1.17 groovy-test 3.0.8

1.17.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. Therefore, please refer to the NOTICE file distributed with the source code.
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/mock/interceptor/package.html

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*  *
*   http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/util/JavadocAssertionTestBuilder.groovy
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/test/GroovyAssert.java
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/mock/interceptor/LooseExpectation.groovy
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/mock/interceptor/GroovyTest Case.java
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/mock/interceptor/StubFor.groovy
* /opt/cola/permits/1218299125_1634929835.17/0/groovy-test-3-0-8-sources-jar/groovy/mock/interceptor/StrictExpectation.groovy
1.18 commons-io 2.5
1.18.1 Available under license:

Apache Commons IO
Copyright 2002-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.19 form-data 1.0.1 1.0.1
1.19.1 Available under license :

Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.20 @babel/highlight 7.16.0
1.20.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.21 promise 2.1.0

1.21.1 Available under license:
Copyright (c) 2014 Forbes Lindesay

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.22 drill-format-ltsv 1.19.0
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.23 curator-x-discovery 5.1.0

1.23.1 Available under license:

Curator Service Discovery
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.24 raw-body 2.3.2 2.3.2

1.24.1 Available under license :
The MIT License (MIT)

Copyright (c) 2013-2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.25 libpng 1.2.54 1ubuntu1.1
1.25.1 Available under license :

This is the pre-packaged Debian Linux version of the libpng graphics library. It was packaged by Philippe Troin <phil@fifi.org> from sources originally retrieved from http://libpng.org/pub/png/libpng.html

The packaging itself is

copyright (C) 2001 Philippe Troin <phil@fifi.org>,
copyright 2002 Junichi Uekawa <dancer@debian.org>,
copyright 2003 Josselin Mouette <joss@debian.org> and
copyright 2006-2009 Anibal Monsalve Salazar <anibal@debian.org>.

It is licensed under the GNU General Public License v2 or later. On Debian systems, the GPL is in /usr/share/common-licenses/GPL-2.

Here is the copyright and license for libpng:

libpng version 1.2.26 - April 2, 2008
Copyright (c) 1998-2008 Glenn Randers-Pehrson
(Version 0.96 Copyright (c) 1996, 1997 Andreas Dilger)
(Version 0.88 Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.)

Authors and maintainers:
libpng versions 0.71, May 1995, through 0.88, January 1996: Guy Schalnat
libpng versions 0.89c, June 1996, through 0.96, May 1997: Andreas Dilger
libpng versions 0.97, January 1998, through 1.2.26 - April 2, 2008: Glenn
See also "Contributing Authors", below.
COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

libpng versions 1.2.6, August 15, 2004, through 1.2.26, April 2, 2008, are
Copyright (c) 2004, 2006-2008 Glenn Randers-Pehrson, and are
distributed according to the same disclaimer and license as libpng-1.2.5
with the following individual added to the list of Contributing Authors:

Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5, October 3, 2002, are
Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are
distributed according to the same disclaimer and license as libpng-1.0.6
with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the
library or against infringement. There is no warranty that our
efforts or the library will fulfill any of your particular purposes
or needs. This library is provided with all faults, and the entire
risk of satisfactory quality, performance, accuracy, and effort is with
the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are
Copyright (c) 1998, 1999, 2000 Glenn Randers-Pehrson, and are
distributed according to the same disclaimer and license as libpng-0.96,
with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996, 1997 Andreas Dilger
Distributed according to the same disclaimer and license as libpng-0.88,
with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, “Contributing Authors”
is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors
and Group 42, Inc. disclaim all warranties, expressed or implied,
including, without limitation, the warranties of merchantability and of
fitness for any purpose. The Contributing Authors and Group 42, Inc.
assume no liability for direct, indirect, incidental, special, exemplary,
or consequential damages, which may result from the use of the PNG
Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this
source code, or portions hereof, for any purpose, without fee, subject
to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and
   must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from
   any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without
fee, and encourage the use of this source code as a component to
supporting the PNG file format in commercial products. If you use this
source code in a product, acknowledgment is not required but would be
appreciated.

---------------------------------------------------------------------------

Copyright (c) 1998-2008 Greg Roelofs. All rights reserved.

This software is provided "as is," without warranty of any kind,
express or implied. In no event shall the author or contributors
be held liable for any damages arising in any way from the use of
this software.
The contents of this file are DUAL-LICENSED. You may modify and/or redistribute this software according to the terms of one of the following two licenses (at your option):

LICENSE 1 ("BSD-like with advertising clause"):

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. Redistributions of source code must retain the above copyright notice, disclaimer, and this list of conditions.
2. Redistributions in binary form must reproduce the above copyright notice, disclaimer, and this list of conditions in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   This product includes software developed by Greg Roelofs and contributors for the book, "PNG: The Definitive Guide," published by O'Reilly and Associates.

LICENSE 2 (GNU GPL v2 or later):

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

---------------------------------------------------------------------------

This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:
If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.

libpng versions 1.0.7, July 1, 2000, through 1.2.54, November 12, 2015, are
Copyright (c) 2000-2002, 2004, 2006-2015 Glenn Randers-Pehrson, are
derived from libpng-1.0.6, and are distributed according to the same
disclaimer and license as libpng-1.0.6 with the following individuals
added to the list of Contributing Authors:

  Simon-Pierre Cadieux
  Eric S. Raymond
  Cosmin Truta
  Gilles Vollant

and with the following additions to the disclaimer:

  There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are
Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

  Tom Lane
  Glenn Randers-Pehrson
  Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

  John Bowler
  Kevin Bracey
  Sam Bushell
  Magnus Holmgren
  Greg Roelofs
  Tom Tanner
libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors"
is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors
and Group 42, Inc. disclaim all warranties, expressed or implied,
including, without limitation, the warranties of merchantability and of
fitness for any purpose. The Contributing Authors and Group 42, Inc.
assume no liability for direct, indirect, incidental, special, exemplary,
or consequential damages, which may result from the use of the PNG
Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this
source code, or portions hereof, for any purpose, without fee, subject
to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not
   be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any
   source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without
fee, and encourage the use of this source code as a component to
supporting the PNG file format in commercial products. If you use this
source code in a product, acknowledgment is not required but would be
appreciated.

END OF COPYRIGHT NOTICE, DISCLAIMER, and LICENSE.

A "png_get_copyright" function is available, for convenient use in "about"
boxes and the like:

    printf("%s", png_get_copyright(NULL));

Also, the PNG logo (in PNG format, of course) is supplied in the
files "pngbar.png" and "pngbar.jpg (88x31) and "pngnow.png" (98x31).
Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative. OSI has not addressed the additional disclaimers inserted at version 1.0.7.

Glenn Randers-Pehrson
glenrp at users.sourceforge.net
November 12, 2015
This is the pre-packaged Debian Linux version of the libpng graphics library. It was packaged by Philippe Troin <phil@fifi.org> from sources originally retrieved from ftp://swrinde.nde.swri.edu/pub/png/src/

The packaging itself is
copyright (C) 2001 Philippe Troin <phil@fifi.org>,
copyright 2002 Junichi Uekawa <dancer@debian.org>,
copyright 2003 Josselin Mouette <joss@debian.org> and
copyright 2006 Anibal Monsalve Salazar <anibal@debian.org>.
It is licensed under the GNU General Public License. On Debian systems, the GPL is in /usr/share/common-licenses/GPL.

Here is the copyright and license for libpng:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:
Open Source Used In DNAC Platform Arica 381

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

1.26 jol-core 0.9
1.26.1 Available under license :

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license
document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share
and change it. By contrast, the GNU General Public License is intended to
guarantee your freedom to share and change free software--to make sure the
software is free for all its users. This General Public License applies to
most of the Free Software Foundation’s software and to any other program whose
authors commit to using it. (Some other Free Software Foundation software is
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included...
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms
of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.27 randomomatic 3.1.1
1.27.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.28 babel-plugin-syntax-async-functions

6.13.0

1.28.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.29 pinkie-promise 1.0.0 1.0.0

1.29.1 Available under license:

The MIT License (MIT)

Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.30 property-expr 2.0.4 2.0.4
1.30.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jason Quense

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.31 websocket-server 9.3.27.v20190418

1.32 babel-plugin-transform-es2015-spread
6.22.0 6.22.0
1.32.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.33 wrap-ansi 7.0.0
1.33.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.34 listencount 1.0.1 1.0.1
1.34.1 Available under license:

Copyright (c) MMXV jden <jason@denizac.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby
granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.

1.35 through 2.0.5

1.35.1 Available under license:

# The MIT License (MIT)

**Copyright (c) Rod Vagg (the "Original Author") and additional contributors**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.36 core-js 2.6.11

1.36.1 Available under license:

Copyright (c) 2014-2019 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.37 unicode-match-property-value-ecmascript 2.0.0

1.37.1 Available under license:
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.38 lodash-repeat 4.1.0 4.1.0

1.38.1 Available under license:
Copyright jQuery Foundation and other contributors <https://jquery.org/> Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>
This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.39 babel-plugin-transform-async-to-generator 6.24.1
1.39.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.40 minimist 1.2.5 1.2.5

1.40.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.41 namespace 1.4.01

1.41.1 Available under license:

xml-commons/java/external/LICENSE.dom-software.txt $Id: LICENSE.dom-software.txt 734314 2009-01-14 03:33:27Z mrglavas $

This license came from: http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.zip (COPYRIGHT.html)

W3C SOFTWARE NOTICE AND LICENSE

All Rights Reserved.

The DOM bindings are published under the W3C Software Copyright Notice and License. The software license requires "Notice of any changes or modifications to the W3C files, including the date changes were made." Consequently, modified versions of the DOM bindings must document that they do not conform to the W3C standard; in the case of the IDL definitions, the pragma prefix can no longer be 'w3c.org'; in the case of the Java language binding, the package names can no longer be in the 'org.w3c' package.

Note: The original version of the W3C Software Copyright Notice and License could be found at http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)
THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

===========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, Version 2.0, in this case for the Apache xml-commons xml-apis distribution. ==
===========================================================================

Apache XML Commons XML APIs
Copyright 1999-2009 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

 Portions of this software were originally based on the following:
 - software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org

This license came from: http://www.w3.org/Consortium/Legal/copyright-documents-20021231

W3C DOCUMENT LICENSE
Public documents on the W3C site are provided by the copyright holders under the following license. By using and/or copying this document, or the W3C document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [$date-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all
This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) $

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

This page is now out of date -- see the new SAX site at http://www.saxproject.org/ for more up-to-date releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

1.42 hbase-procedure 2.4.2
1.42.1 Available under license:

Apache HBase - Procedure
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.43 babel-plugin-transform-strict-mode

6.24.1 6.24.1

1.43.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.44 etag 1.7.0 1.7.0
1.44.1 Available under license:
(The MIT License)

Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.45 composite-error 0.1.1 0.1.1

1.45.1 Available under license:
(The MIT License)

Copyright (c) 2015 Nick Uraltsev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.

1.46 is-my-json-valid 2.20.5
1.46.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.47 lodash 4.17.15 4.17.15
1.47.1 Available under license :
The MIT License (MIT)

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.48 @babel/plugin-syntax-optional-catch-binding 7.8.3
1.48.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.49 grpc-api 1.27.1

1.49.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the
Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by
brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and
description of purpose be included on the same "printed page" as the copyright notice for easier identification within
third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.50 json3 3.3.2 3.3.2

1.50.1 Available under license:
Copyright (c) 2012-2014 Kit Cambridge.
http://kitcambridge.be/

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.51 jetty-xml 9.4.41.v20210516

1.51.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.
"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.
For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient’s rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient’s rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient’s obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

=================================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.
* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.52 Iru-cache 4.1.1 4.1.1

1.52.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.53 groovy-groovydoc 3.0.8

1.53.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyDocTemplateEngine.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyRootDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyDocParser.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyRootDocBuilder.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/LinkArgument.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyAbstractableElementDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/ExternalGroovyClassDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyConstructorDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/FileOutputTool.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/groovydoc/GroovyConstructorDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyFieldDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyExecutableMemberDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/ResourceManager.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyProgramElementDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/groovydoc/GroovyRootDoc.java
No license file was found, but licenses were detected in source scan.

/*! normalize.css v2.1.0 | MIT License | git.io/normalize */

Found in path(s):
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/apache/groovy/groovydoc/tools/GroovyDocUtil.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/groovydoc/GroovyClassDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyDocWriter.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/Main.groovy
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyMethodDoc.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/GroovyDocWriter.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/Main.groovy
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyAnnotationRef.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/ClasspathResourceManager.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/FileSystemResourceManager.java
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/SimpleGroovyClassDocAssembler.java
No license file was found, but licenses were detected in source scan.

LICENSE

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file
distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/allclasses-frame.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/package.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/packageLevel/package-frame.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/packageLevel/packageDocStructuredData.xml
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/index-all.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/help-doc.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovydoc/package.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/index.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/classLevel/classDocStructuredData.xml
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/classLevel/classDocName.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/packageLevel/package-summary.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/deprecated-list.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/overview-summary.html
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/rootDocStructuredData.xml
* /opt/cola/permits/1218304752_1634925093.2/0/groovy-groovydoc-3-0-8-sources-jar/org/codehaus/groovy/tools/groovydoc/gstringTemplates/topLevel/overview-frame.html
1.54 qs 6.10.1 6.10.1
1.54.1 Available under license:

BSD 3-Clause License

Copyright (c) 2014, Nathan LaFreniere and other [contributors](https://github.com/ljharb/qs/graphs/contributors)
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.55 caniuse-lite 1.0.30001282
1.55.1 Available under license:

Attribution 4.0 International

================================================================================
Creative Commons Corporation ("Creative Commons") is not a law firm and
does not provide legal services or legal advice. Distribution of
Creative Commons public licenses does not create a lawyer-client or
other relationship. Creative Commons makes its licenses and related
information available on an "as-is" basis. Creative Commons gives no
warranties regarding its licenses, any material licensed under their
terms and conditions, or any related information. Creative Commons
disclaims all liability for damages resulting from their use to the
fullest extent possible.

Using Creative Commons Public Licenses

Creative Commons public licenses provide a standard set of terms and
conditions that creators and other rights holders may use to share
original works of authorship and other material subject to copyright
and certain other rights specified in the public license below. The
following considerations are for informational purposes only, are not
exhaustive, and do not form part of our licenses.

Considerations for licensors: Our public licenses are
intended for use by those authorized to give the public
permission to use material in ways otherwise restricted by
copyright and certain other rights. Our licenses are
irrevocable. Licensors should read and understand the terms
and conditions of the license they choose before applying it.
Licensors should also secure all rights necessary before
applying our licenses so that the public can reuse the
material as expected. Licensors should clearly mark any
material not subject to the license. This includes other CC-
licensed material, or material used under an exception or
limitation to copyright. More considerations for licensors:
wiki.creativecommons.org/Considerations_for_licensors

Considerations for the public: By using one of our public
licenses, a licensor grants the public permission to use the
licensed material under specified terms and conditions. If
the licensor's permission is not necessary for any reason—for
example, because of any applicable exception or limitation to
copyright—then that use is not regulated by the license. Our
licenses grant only permissions under copyright and certain
other rights that a licensor has authority to grant. Use of
the licensed material may still be restricted for other
reasons, including because others have copyright or other
rights in the material. A licensor may make special requests,
such as asking that all changes be marked or described.
Although not required by our licenses, you are encouraged to
respect those requests where reasonable. More_considerations
for the public:
wiki.creativecommons.org/Considerations_for_licensees

=======================================================================

Creative Commons Attribution 4.0 International Public License
By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License (“Public License”). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 -- Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.

b. Adapter's License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

f. Licensed Material means the artistic or literary work, database, or other material to which the Licensor applied this Public License.
g. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. Licensor means the individual(s) or entity(ies) granting rights under this Public License.

i. Share means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

k. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

Section 2 -- Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

   a. reproduce and Share the Licensed Material, in whole or in part; and

   b. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).
4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, simply making modifications authorized by this Section 2(a) (4) never produces Adapted Material.

5. Downstream recipients.

a. Offer from the Licensor -- Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

b. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.
3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 -- License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

1. If You Share the Licensed Material (including in modified form), You must:

a. retain the following if it is supplied by the Licensor with the Licensed Material:

i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);

ii. a copyright notice;

iii. a notice that refers to this Public License;

iv. a notice that refers to the disclaimer of warranties;

v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;

b. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and

c. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.

2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or
3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.

4. If You Share Adapted Material You produce, the Adapter's License You apply must not prevent recipients of the Adapted Material from complying with this Public License.

Section 4 -- Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;

b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and

c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 -- Disclaimer of Warranties and Limitation of Liability.

a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE. WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.

b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE
TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.

c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 -- Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:

1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

2. upon express reinstatement by the Licensor.

For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 -- Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.
b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 -- Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.

d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

=======================================================================

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the Licensor. The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.
1.56 jackson-datatype-joda 2.4.2
1.56.1 Available under license:
This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.
You may obtain a copy of the License at:
http://www.apache.org/licenses/LICENSE-2.0

1.57 options 0.0.6 0.0.6
1.57.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.58 path-exists 4.0.0
1.58.1 Available under license:
MIT License
Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.59 swagger-jaxrs2 2.0.7

1.59.1 Available under license:

Copyright 2018 SmartBear Software

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at [apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.60 @babel/helper-optimise-call-expression

7.16.0

1.60.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.61 abbrev 1.1.1 1.1.1
1.61.1 Available under license:

This software is dual-licensed under the ISC and MIT licenses.
You may use this software under EITHER of the following licenses.

--------

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

--------

Copyright Isaac Z. Schlueter and Contributors
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.62 kind-of 3.2.2 3.2.2
1.62.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.63 map-age-cleaner 0.1.3 0.1.3
1.63.1 Available under license :
   MIT License

Copyright (c) Sam Verschueren <sam.verschueren@gmail.com> (github.com/SamVerschueren)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.64 stax 4.2
1.64.1 Available under license :

This copy of Stax2 API is licensed under the
Simplified BSD License (also known as "2-clause BSD", or "FreeBSD License")
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.opensource.org/licenses/bsd-license.php

with details of:

<COPYRIGHT HOLDER> = FasterXML.com
<YEAR> = 2010-

1.65 grpc-context 1.27.1
1.65.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.66 ansi-regex 5.0.0 5.0.0

1.66.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.67 toggle-selection 1.0.6 1.0.6

1.67.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.68 es6-symbol 3.1.3

1.68.1 Available under license :

ISC License

Copyright (c) 2013-2019, Mariusz Nowak, @medikoo, medikoo.com

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.69 jsonpointer 4.1.0

1.69.1 Available under license :

The MIT License (MIT)

Copyright (c) 2011-2015 Jan Lehnardt <jan@apache.org> & Marc Bachmann <https://github.com/marcbachmann>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.70 tiny-inflate 1.0.3 1.0.3

1.70.1 Available under license:
MIT License

Copyright (c) 2015-present Devon Govett

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.71 drill-kudu-storage 1.19.0

1.71.1 Available under license:

Drill : Contrib : Storage : Kudu
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.72 yargs-unparser 1.6.0

1.72.1 Available under license:

The MIT License (MIT)

Copyright (c) 2017 Made With MOXY Lda <hello@moxy.studio>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.73 source-map 0.5.7 0.5.7

1.73.1 Available under license:

Copyright (c) 2009-2011, Mozilla Foundation and contributors All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.74 parquet-avro 1.8.1
1.74.1 Available under license:

Apache Parquet Avro
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.75 drill-storage-hive-core 1.19.0
1.75.1 Available under license :

Drill : Contrib : Storage : Hive : Core
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
1.76 oas-resolver 2.5.4 2.5.4
1.76.1 Available under license:
BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.77 resolve-dir 1.0.1 1.0.1
1.77.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.78 find-cache-dir 3.3.2
1.78.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.79 helmet-csp 1.2.2 1.2.2
1.79.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2016 Evan Hahn, Adam Baldwin
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.80 helmet 2.3.0 2.3.0

1.80.1 Available under license :
(The MIT License)

Copyright (c) 2012-2016 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.81 lodash.flow 3.5.0 3.5.0

1.81.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.82 yargs 15.4.1

1.82.1 Available under license:
MIT License

Copyright 2010 James Halliday (mail@substack.net); Modified work Copyright 2014 Contributors (ben@npmjs.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.83 form-data 2.3.3 2.3.3

1.83.1 Available under license:
Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.84 expiringmap 0.5.8

1.85 is-yarn-global 0.3.0

1.85.1 Available under license:

MIT License

Copyright (c) 2018 LitoMore

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.86 checker-qual 3.5.0

1.86.1 Available under license:

Checker Framework qualifiers

Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.87 ansi-regex 4.1.0 4.1.0
1.87.1 Available under license :
  MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.88 ms 2.1.2
1.88.1 Available under license :
  The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.89 json-path 1.1.1 1.1.1

1.89.1 Available under license:

Copyright (c) 2014-2016 David Chester <david@fmail.co.uk>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.90 hibernate-validator 5.0.0.Final

1.90.1 Available under license:

Found license 'GNU Lesser General Public License' in * This is free software; you can redistribute it and/or modify it * under the terms of the GNU Lesser General Public License as * published by the Free Software Foundation; either version 2.1 of * the License, or (at your option) any later version. * This software is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details. * You should have received a copy of the GNU Lesser General Public * Expert Group and
1.91 jersey-media-multipart 2.5.1

1.91.1 Available under license:
CDDL+GPL License
http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.92 widest-line 3.1.0

1.92.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.93 is-fullwidth-code-point 3.0.0 3.0.0

1.93.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.94 netty-transport 4.1.42.Final
1.94.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/PendingWriteQueue.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelFactory.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/embedded/EmbeddedChannelId.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/AbstractEventLoop.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ReflectiveChannelFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
/**
 * Handles an I/O event or intercepts an I/O operation, and forwards it to its next handler in its \{link ChannelPipeline\}.
 */

<h3>Sub-types</h3>

* {@link ChannelInboundHandler} to handle inbound I/O events, and</li>
* {@link ChannelOutboundHandler} to handle outbound I/O operations.
</li>
</ul>

Alternatively, the following adapter classes are provided for your convenience:

* {@link ChannelInboundHandlerAdapter} to handle inbound I/O events,
* {@link ChannelOutboundHandlerAdapter} to handle outbound I/O operations,
* {@link ChannelDuplexHandler} to handle both inbound and outbound events

For more information, please refer to the documentation of each subtype.

<h3>The context object</h3>

A {@link ChannelHandler} is provided with a {@link ChannelHandlerContext} object. A {@link ChannelHandler} is supposed to interact with the {@link ChannelPipeline} it belongs to via a context object. Using the context object, the {@link ChannelHandler} can pass events upstream or downstream, modify the pipeline dynamically, or store the information (using {@link AttributeKey}s) which is specific to the handler.

<h3>State management</h3>

A {@link ChannelHandler} often needs to store some stateful information. The simplest and recommended approach is to use member variables:
public interface Message {
    // your methods here
}

public class DataServerHandler extends SimpleChannelInboundHandler<Message> {
    private boolean loggedIn;

    @Override
    public void channelRead0(ChannelHandlerContext ctx, Message message) {
        if (message instanceof LoginMessage) {
            authenticate((LoginMessage) message);
            loggedIn = true;
        } else if (message instanceof GetDataMessage) {
            if (loggedIn) {
                ctx.writeAndFlush(fetchSecret((GetDataMessage) message));
            } else {
                fail();
            }
        }
    }

    // Create a new handler instance per channel.
    // See ChannelInitializer.initChannel(Channel).
    public class DataServerInitializer extends ChannelInitializer<Channel> {
        @Override
        public void initChannel(Channel channel) {
            channel.pipeline().addLast("handler", new DataServerHandler());
        }
    }

    // Using AttributeKey
    // Although it's recommended to use member variables to store the state of a
    // handler, for some reason you might not want to create many handler instances.
    // In such a case, you can use AttributeKey which is provided by
    public interface Message {
* // your methods here
* }
*
* {@code @Sharable}
* public class DataServerHandler extends SimpleChannelInboundHandler
* {
*     private final AttributeKey&lt;Boolean&gt; auth = AttributeKey.valueOf("auth");
*     {
*         auth.set(true);
*     }
*     {
*         try {
*             authenticate((LoginMessage) o);
*             attr.set(true);
*         } catch (RuntimeException e) {
*             fail();
*         }
*         ...
*     }
* }
* </pre>
* Now that the state of the handler is attached to the ChannelHandlerContext, you can add the same handler instance to different pipelines:
* <pre>
* public class DataServerInitializer extends ChannelInitializer
* {
*     private static final DataServerHandler SHARED = new DataServerHandler();
*     {
*         channel.pipeline().addLast("handler", SHARED);
*     }
* }
* </pre>
*<h4>The @Sharable annotation</h4>
* In the example above which used an AttributeKey, you might have noticed the @Sharable annotation. If a ChannelHandler is annotated with the @Sharable annotation, it means you can create an instance of the handler just once and add it to one or more ChannelPipeline's multiple times without
* a race condition.
* <p>
* If this annotation is not specified, you have to create a new handler
* instance every time you add it to a pipeline because it has unshared state
* such as member variables.
* <p>
* This annotation is provided for documentation purpose, just like
* <a href="http://www.javaconcurrencyinpractice.com/annotations/doc/">the JCIP annotations</a>
* .
* <h3>Additional resources worth reading</h3>
* <p>
* Please refer to the [@link ChannelHandler], and
* [@link ChannelPipeline] to find out more about inbound and outbound operations,
* what fundamental differences they have, how they flow in a pipeline, and how to handle
* the operation in your application.
*/

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/ChannelHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/AbstractCoalescingBufferQueue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/CoalescingBufferQueue.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/nio/NioEventLoopGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/nio/NioEventLoop.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/nio/NioEventLoopFutureListener.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/nio/nio/NioDatagramChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/nio/DefaultEventLoopGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/socket/DatagramChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/group/CombinedIterator.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/socket/DefaultEventLoopGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/socket/nio/NioDatagramChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
Open Source Used In DNAC Platform Arica 487

jar/io/netty/channel/oio/AbstractOioMessageChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/local/LocalServerChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/package-info.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/local/LocalChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/InternetProtocolFamily.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/group/DefaultChannelGroupFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/ServerSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/DatagramChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/CompleteChannelFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/ServerBootstrap.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/AbstractServerChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ThreadPerChannelEventLoop.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/package-info.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelInitializer.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelMetadata.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/EventLoop.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/nio/package-info.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/ProtocolFamilyConverter.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/DefaultChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/nio/ProtocolFamilyConverter.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/nio/DefaultChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/NioServerSocketChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/SocketChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/SingleThreadEventLoop.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
Open Source Used In DNAC Platform Arica 488
jar/io/netty/channel/ChannelInboundHandlerAdapter.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/local/LocalAddress.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/socket/package-info.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/DefaultChannelPromise.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/socket/DefaultServerSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/oio/AbstractOioChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/local/LocalChannelRegistry.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/socket/nio/NioSocketChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/multiplex/MultiplexChannelGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/EventLoopGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/FailedChannelFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/socket/oio/OioServerSocketChannel.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 */
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/MaxBytesRecvByteBufAllocator.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultMaxMessagesRecvByteBufAllocator.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/ChannelPoolMap.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/AbstractChannelPoolHandler.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/SimpleChannelPool.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/FixedChannelPool.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/ChannelHealthChecker.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/AbstractChannelPoolMap.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultMaxBytesRecvByteBufAllocator.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/MaxMessagesRecvByteBufAllocator.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/ChannelPool.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/package-info.java
  * /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/pool/ChannelPoolHandler.java

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
#   http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

/*
* Found in path(s):
* No license file was found, but licenses were detected in source scan.
*/

/*
* Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/group/ChannelGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/ChannelProgressivePromise.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/ChannelProgressiveFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/ChannelProgressiveFutureListener.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/socket/oio/DefaultOioSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/socket/oio/OioServerSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/nio/SelectedSelectionKeySet.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io.netty/channel/group/DefaultChannelGroup.java
*/
Open Source Used In DNAC Platform Arica  493

* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultChannelProgressivePromise.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/AbstractEventLoopGroup.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultChannelId.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/OioSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ConnectTimeoutException.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/DefaultOioServerSocketChannelConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelMatchers.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/group/ChannelGroupException.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultMessageSizeEstimator.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultAddressedEnvelope.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelHandlerAdapter.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/MessageSizeEstimator.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/ChannelFactory.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2019 The Netty Project
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/EventLoopTaskQueueFactory.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/DefaultChannelHandlerContext.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/socket/nio/NioChannelOption.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
jar/io/netty/channel/SimpleUserEventChannelHandler.java
Open Source Used In DNAC Platform Arica

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/AbstractBootstrapConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultSelectStrategy.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/group/VoidChannelGroupFuture.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/SelectStrategyFactory.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DefaultSelectStrategyFactory.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelInboundInvoker.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/ServerBootstrapConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/ChannelOutboundInvoker.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/WriteBufferWaterMark.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/SelectStrategy.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/PreferHeapByteBufAllocator.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/BootstrapConfig.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/DuplexChannel.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/META-INF/maven/io.netty/netty-transport/pom.xml
  No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 */

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/ChannelInputShutdownReadComplete.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/internal/ChannelUtils.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/bootstrap/FailedChannel.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/ChannelOutputShutdownEvent.java
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-jar/io/netty/channel/socket/oio/DefaultOioDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272150_1643050038.76/0/netty-transport-4-1-42-final-sources-
  jar/io/netty/channel/oio/OioByteStreamChannel.java

1.95 image-size 0.7.4 0.7.4

1.95.1 Available under license :

The MIT License (MIT)

Copyright 2017 Aditya Yadav, http://netroy.in

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the Software), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.96 archiver 2.1.1 2.1.1

1.96.1 Available under license:

Copyright (c) 2012-2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.97 get-intrinsic 1.1.1

1.97.1 Available under license:

MIT License

Copyright (c) 2020 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.98 pcapngdecoder 1.2

1.98.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * The MIT License (MIT)
 * <p/>
 * Copyright (c) 2015-2016 Bertrand Martel
 * <p/>
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 * <p/>
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 * <p/>
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 * THE SOFTWARE.
 */

Found in path(s):
* /opt/cola/permits/1202446644_1631567950.01/0/pcapngdecoder-1-2-sources-jar/fr/bmartel/pcapdecoder/structure/options/implementations/OptionsEnhancedHeader.java
* /opt/cola/permits/1202446644_1631567950.01/0/pcapngdecoder-1-2-sources-jar/fr/bmartel/pcapdecoder/structure/types/implementations/NameResolutionHeader.java
* /opt/cola/permits/1202446644_1631567950.01/0/pcapngdecoder-1-2-sources-jar/fr/bmartel/pcapdecoder/structure/types/implementations/NameResolutionHeader.java

Open Source Used In DNAC Platform Arica 499
1.99 firstline 1.3.1 1.3.1

1.99.1 Available under license:
The MIT License (MIT)

Copyright (c) 2018 Alessandro Zanardi

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.100 grpc-stub 1.27.1

1.100.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent
infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.101 batik-dom 1.13

1.101.1 Available under license:

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (c) 2005 World Wide Web Consortium,
* (Massachusetts Institute of Technology, European Research Consortium for
* Informatics and Mathematics, Keio University). All Rights Reserved. This
* work is distributed under the W3C(r) Software License [1] in the hope that
* it will be useful, but WITHOUT ANY WARRANTY; without even the implied
* warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
* Modifications:
* September 10, 2005
* Placed interface in org.apache.batik.dom.xbl for the time being.
* Added javadocs.
*/

Found in path(s):
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1.jar/org/apache/batik/dom/xbl/ShadowTreeEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1.jar/org/apache/batik/dom/xbl/NodeXBL.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1.jar/org/apache/batik/dom/xbl/XBLShadowTreeElement.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1.jar/org/apache/batik/dom/xbl/OriginalEvent.java

No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericProcessingInstruction.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/events/DOMKeyboardEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/ExtendedNode.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/events/DOMMouseEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/AbstractCharacterData.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/events/DOMMutationNameEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/StyleSheetProcessingInstruction.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericNotation.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/ExtensibleDOMImplementation.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/AbstractElement.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/events/DOMMutationNameEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/StyleSheetProcessingInstruction.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericNotation.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/ExtensibleDOMImplementation.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericDocument.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-
jar/org/apache/batik/dom/AbstractParentNode.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/events/DOMTextEvent.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/AbstractNode.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/DomExtension.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/util/DOMUtilities.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericComment.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/AbstractText.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/util/HashTableStack.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/GenericAttr.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/AbstractEntity.java
* /opt/cola/permits/1146878549_1616835234.92/0/batik-dom-1-13-sources-1-jar/org/apache/batik/dom/util/SAXDocumentFactory.java

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership.

The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
1.102 referrer-policy 1.0.0 1.0.0
1.102.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Evan Hahn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.103 commons-net 1.4.1
1.103.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2005 The Apache Software Foundation
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Open Source Used In DNAC Platform Arica

Found in path(s):
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/smtp/SMTPCommand.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/parser/VMSFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/FTPFileEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/FTPFileIterator.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/TFTPFile.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/UnixFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/DefaultDatagramSocketFactory.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/ProtocolCommandEvent.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/TFTPWriteRequestPacket.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/io/SocketInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/pop3/POP3Reply.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/FTPFileListParserImpl.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/FTPFileList.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/DatagramSocketFactory.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/parser/VMSVersioningFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/SMTPReply.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/nntp/SimpleNNTPHeader.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/ftp/FromNetASCIIOutputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/tftp/TFTPRequestPacket.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1.4.1-sources-1.jar/org/apache/commons/net/smtp/SMTPClient.java
jar/org/apache/commons/net/ntp/NtpUtils.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPFile.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/TFTPErrorPacket.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/DotTerminatedMessageWriter.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/CharGenUDPClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/pop3/POP3Client.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/bsd/RExecClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/EchoTCPClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/TimeTCPClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/smtp/SMTPConnectionClosedException.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ProtocolCommandListener.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/FromNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/smtp/SMTP.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIOutputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/newt/NewsgroupInfo.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/bsd/RLoginClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/parser/EnterpriseUnixFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/SocketFactory.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/parser/NTFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/bsd/RLoginClient.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPConnectionClosedException.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPReply.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/smtp/RelayPath.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/parser/EnterpriseUnixFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/util/ListenerList.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTP.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPFile.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPReply.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/smtp/RelayPath.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/parser/EnterpriseUnixFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/util/ListenerList.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTP.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTPReply.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/io/ToNetASCIIInputStream.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/smtp/RelayPath.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/parser/EnterpriseUnixFTPEntryParser.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/util/ListenerList.java
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
jar/org/apache/commons/net/ftp/FTP.java
jar/org/apache/commons/net/nntp/NNTPCommand.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/ntp/TimeInfo.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/EchoUDPClient.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/nntp/ArticlePointer.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2003-2004 The Apache Software Foundation
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/SimpleOptionHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/TelnetNotificationHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/SuppressGAOptionHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/InvalidTelnetOptionException.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/TelnetOptionHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/TelnetInputStream.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/TerminalTypeOptionHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/EchoOptionHandler.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/TelnetClient.java
* /opt/cola/permits/113647281_1613772384.01/0/commons-net-1-4-1-sources-1-jar/org/apache/commons/net/telnet/Telnet.java

No license file was found, but licenses were detected in source scan.
/* ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2001-2005 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *       "This product includes software developed by the
 *       Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation" and
 * "Apache Commons" must not be used to endorse or promote products
 * derived from this software without prior written permission. For
 * written permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache",
 *    nor may "Apache" appear in their name, without
 *    prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * ====================================================================
 * */
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

No license file was found, but licenses were detected in source scan.

* Copyright 2005 The Apache Software Foundation

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

No license file was found, but licenses were detected in source scan.

* Copyright 2001-2005 The Apache Software Foundation

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/***
 * List all new newsgroups added to the NNTP server since a particular
 * date subject to the conditions of the specified query. If no new
 * newsgroups were added, a zero length array will be returned. If the
 * command fails, null will be returned.
 * <p>
 * @param query  The query restricting how to search for new newsgroups.
 * @return An array of NewsgroupInfo instances containing the information
 * for each new newsgroup added to the NNTP server. If no newsgroups
 * were added, a zero length array will be returned. If the command
 * fails, null will be returned.
 * @exception NNTPConnectionClosedException
 *   If the NNTP server prematurely closes the connection as a result
 *   of the client being idle or some other reason causing the server
 *   to send NNTP reply code 400. This exception may be caught either
 *   as an IOException or independently as itself.
 * @exception IOException  If an I/O error occurs while either sending a
 *   command to the server or receiving a reply from the server.
 ***/

Found in path(s):
* /opt/cola/permits/1136474281_1613772384.01/0/commons-net-1-4-1-sources-1-
  jar/org/apache/commons/net/nntp/NNTPClient.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2004 The Apache Software Foundation
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Open Source Used In DNAC Platform Arica  520

1.104 inquirer 0.12.0

1.104.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed under the MIT license.

Found in path(s):
* /opt/cola/permits/11612566323_1620862018.21/0/inquirer-0-12-0-4-tgz/package/README.md
1.105 engine.io 3.4.2 3.4.2

1.105.1 Available under license:
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.106 ipaddr.js 1.0.5 1.0.5

1.106.1 Available under license:
Copyright (C) 2011 Peter Zotov <whitequark@whitequark.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.107 destroy 1.0.4 1.0.4
1.107.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.108 hive-storage-api 2.7.0
1.108.1 Available under license:

Hive Storage API
Copyright 2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attributions notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.109 is-glob 4.0.1

1.109.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.110 istanbul-reports 3.0.2

1.110.1 Available under license:
Copyright 2012-2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.111 xsdlib 2013.6.1

1.111.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2001-2013 Oracle and/or its affiliates. All rights reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 
 * - Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 * 
 * - Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in the
 *   documentation and/or other materials provided with the distribution.
 * 
 * - Neither the name of Oracle nor the names of its
 *   contributors may be used to endorse or promote products derived
 *   from this software without specific prior written permission.
 * 
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
 * IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
 * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
 * PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
 * LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
 * NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */
Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/LongType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/BuiltinAtomicType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/XSDatatypeImpl.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/ListType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/xml/util/XmlChars.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/Base64BinaryType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/GYearMonthType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/IntegerDerivedType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MinLengthFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/TypeIncubator.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/DurationType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/TokenType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MaxExclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/UnionType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/FinalComponent.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MaxLengthFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MinLengthFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/NormalizedStringType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/HexBinaryType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/UnsignedLongType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MinLengthFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MaxExclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/UnionType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/FinalComponent.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MaxExclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/UnionType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MinExclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/MinLengthFacet.java
Open Source Used In DNAC Platform Arica 530

jar/com/sun/msv/datatype/xsd/EnumerationFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/datetime/PreciseCalendarParser.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/BooleanType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/MaxInclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/NegativeIntegerType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/datetime/IDateTimeValueType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/DateTimeFactory.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/AbstractCalendarFormatter.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/BigDateTimeValueType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/Util.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/NumberType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/WhiteSpaceProcessor.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/MinInclusiveFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/DoubleType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/XmlNames.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/ByteType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/WhiteSpaceFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/DataTypeWithLexicalConstraintFacet.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/DatatypeFactory.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/ITimeDurationValueType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/QnameValueType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/NcnameType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/xsd/StringType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/DbatbindableDatatype.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
Open Source Used In DNAC Platform Arica

```
jar/com/sun/msv/datatype/xsd/datetime/BigTimeDurationValueType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/NonNegativeIntegerType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/datetime/CalendarParser.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/GDayType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/IntegerType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/IDType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/QnameType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/BinaryBaseType.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
jar/com/sun/msv/datatype/xsd/IDREFType.java

No license file was found, but licenses were detected in source scan.

/*

* The Apache Software License, Version 1.1
*
*
* Copyright (c) 1999-2003 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Xerces" and "Apache Software Foundation" must
*    not be used to endorse or promote products derived from this
*    software without prior written permission. For written
```
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache",
nor may "Apache" appear in their name, without prior written
permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation and was
originally based on software copyright (c) 1999, International
Business Machines, Inc., http://www.apache.org. For more
information on the Apache Software Foundation, please see

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
Neither the name of Oracle nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/package.html

No license file was found, but licenses were detected in source scan.

/*============================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 1999-2013 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* 
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Xerces" and "Apache Software Foundation" must
*    not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
/*============================================================================
* 5. Products derived from this software may not be called "Apache",
* nor may "Apache" appear in their name, without prior written
* permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999, International
* Business Machines, Inc., http://www.apache.org. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/RangeToken.java
No license file was found, but licenses were detected in source scan.

# Copyright (c) 2001-2013 Oracle and/or its affiliates. All rights reserved.
# modification, are permitted provided that the following conditions are
# - Redistributions of source code must retain the above copyright
# notice, this list of conditions and the following disclaimer.
# - Redistributions in binary form must reproduce the above copyright
# notice, this list of conditions and the following disclaimer in the
# documentation and/or other materials provided with the distribution.
# this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/message.properties
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/message_ja.properties
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/message_fr.properties
No license file was found, but licenses were detected in source scan.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 * 
 * Copyright (c) 2001-2013 Oracle and/or its affiliates. All rights reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are
 * met:
 * 
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 
 * - Redistribution in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 
 * Neither the name of Sun Microsystems, Inc. or the names of
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
 * 
 * This software is provided "AS IS," without a warranty of any kind. ALL
 * EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES,
 * INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A
 * PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND
 * ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES OR LIABILITIES
 * SUFFERED BY LICENSEE AS A RESULT OF OR RELATING TO USE, MODIFICATION
 * OR DISTRIBUTION OF THE SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL
 * SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA,
 * OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR
 * PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF
 * LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE,
 * EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
 */

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-
  jar/com/sun/msv/datatype/regexp/InternalImpl.java

No license file was found, but licenses were detected in source scan.

/*
 * The Apache Software License, Version 1.1
 * 
 * * Copyright (c) 1999-2002 The Apache Software Foundation. All rights
 * reserved.
 * 
 * */
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Xerces" and "Apache Software Foundation" must
* not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
* nor may "Apache" appear in their name, without prior written
* permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999, International
* Business Machines, Inc., http://www.apache.org. For more
* information on the Apache Software Foundation, please see
/*

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/BMPattern.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/RegularExpression.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/ParseException.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/Match.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/REUtil.java
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/regexp/Token.java

No license file was found, but licenses were detected in source scan.

# Copyright (c) 2001-2013 Oracle and/or its affiliates. All rights reserved.
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
# - Redistributions of source code must retain the above copyright
#   notice, this list of conditions and the following disclaimer.
# - Redistributions in binary form must reproduce the above copyright
#   notice, this list of conditions and the following disclaimer in the
#   documentation and/or other materials provided with the distribution.
# - Neither the name of Oracle nor the names of its
#   from this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/Messages.properties
* /opt/cola/permits/1257217844_1642790354.82/0/xsdlib-2013-6-1-sources-jar/com/sun/msv/datatype/xsd/Messages_ja.properties

1.112 jetty-jmx 9.4.34.v20201102
1.112.1 Available under license :

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is
added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of
6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program
(including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core
The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------
Apache
The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.
1.113 nanoid 1.0.2 1.0.2

1.113.1 Available under license:

The MIT License (MIT)

Copyright 2017 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.114 os-locale 3.1.0 3.1.0

1.114.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.115 @babel/helper-function-name 7.16.0
1.115.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.116 woodstox-core 6.2.0
1.116.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.117 es6-error 4.1.1
1.117.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Ben Youngblood
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.118 hbase-shaded-jetty 3.4.1

1.118.1 Available under license:

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

=================================================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=================================================================================================

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache
The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.119 jsesc 0.5.0 0.5.0
Available under license:

Copyright Mathias Bynens <http://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libjpeg-turbo 6b

Available under license:

libjpeg-turbo Licenses

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

- The IJG (Independent JPEG Group) License, which is listed in [README.ijg](README.ijg)

  This license applies to the libjpeg API library and associated programs (any code inherited from libjpeg, and any modifications to that code.)

- The Modified (3-clause) BSD License, which is listed below

  This license covers the TurboJPEG API library and associated programs, as well as the build system.

- The [zlib License](https://opensource.org/licenses/Zlib)

  This license is a subset of the other two, and it covers the libjpeg-turbo SIMD extensions.
Complying with the libjpeg-turbo Licenses

This section provides a roll-up of the libjpeg-turbo licensing terms, to the best of our understanding.

1. If you are distributing a modified version of the libjpeg-turbo source, then:

1. You cannot alter or remove any existing copyright or license notices from the source.

**Origin**
- Clause 1 of the IJG License
- Clause 1 of the Modified BSD License
- Clauses 1 and 3 of the zlib License

2. You must add your own copyright notice to the header of each source file you modified, so others can tell that you modified that file (if there is not an existing copyright header in that file, then you can simply add a notice stating that you modified the file.)

**Origin**
- Clause 1 of the IJG License
- Clause 2 of the zlib License

3. You must include the IJG README file, and you must not alter any of the copyright or license text in that file.

**Origin**
- Clause 1 of the IJG License

2. If you are distributing only libjpeg-turbo binaries without the source, or if you are distributing an application that statically links with libjpeg-turbo, then:

1. Your product documentation must include a message stating:

This software is based in part on the work of the Independent JPEG Group.

**Origin**
- Clause 2 of the IJG license

2. If your binary distribution includes or uses the TurboJPEG API, then your product documentation must include the text of the Modified BSD License (see below.)
3. You cannot use the name of the IJG or The libjpeg-turbo Project or the contributors thereof in advertising, publicity, etc.

4. The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be free of defects, nor do we accept any liability for undesirable consequences resulting from your use of the software.

The Modified (3-clause) BSD License
====================================

Copyright (C)2009-2019 D. R. Commander. All Rights Reserved.
Copyright (C)2015 Viktor Szathmry. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS", AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Why Three Licenses?
=================================

The zlib License could have been used instead of the Modified (3-clause) BSD License, and since the IJG License effectively subsumes the distribution conditions of the zlib License, this would have effectively placed libjpeg-turbo binary distributions under the IJG License. However, the IJG License specifically refers to the Independent JPEG Group and does not extend attribution and endorsement protections to other entities. Thus, it was desirable to choose a license that granted us the same protections for new code that were granted to the IJG for code derived from their software.

1.121 netty 4.1.59.Final
1.121.1 Available under license :
Copyright (c) 2003-2008 Yuta Mori All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is
described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or
 Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
 stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
 that You distribute, all copyright, patent, trademark, and
 attribution notices from the Source form of the Work,
 excluding those notices that do not pertain to any part of
 the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
 names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The MIT License

Copyright (c) 2009 William Kinney

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright 2009-2010 Ning, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 * Copyright (c) 2004-2006, 2008, 2009, 2011 Apple Inc. All rights reserved.
 */

* @APPLE_LICENSE_HEADER_START@
*
* * This file contains Original Code and/or Modifications of Original Code
* as defined in and that are subject to the Apple Public Source License
* Version 2.0 (the 'License'). You may not use this file except in
* compliance with the License. Please obtain a copy of the License at
* https://www.opensource.apple.com/apsl/ and read it before using this
* file.
* *
* * The Original Code and all software distributed under the License are
* * distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER
* * EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010-2011 Matthew J. Francis and Contributors of the jbzip2 Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicatees holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.
Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Netty Project

Please visit the Netty web site for more information:

* https://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations
under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

-------------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * https://www.slf4j.org/

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* NOTICE:
  * license/NOTICE.harmony.txt
* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
* https://archive.apache.org/dist/harmony/

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://github.com/y-256/libdivsufsort

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
  * HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
  * HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
  * HOMEPAGE:
  * https://github.com/google/protobuf

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
  * HOMEPAGE:
  * https://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
  * HOMEPAGE:
  * https://github.com/google/snappy

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/caliper

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * https://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.alto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Cory Benfield. It can be obtained at:

* LICENSE:
This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Tatsuhiro Tsujikawa. It can be obtained at:

* LICENSE:
  * license/LICENSE.nghttp2-hpack.txt (MIT License)
  * HOMEPAGE:
  * https://github.com/nghttp2/nghttp2/

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
  * HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:
  * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)
  * HOMEPAGE:
  * https://github.com/takari/maven-wrapper

This product contains the dnsinfo.h header file, that provides a way to retrieve the system DNS configuration on MacOS.
This private header is also used by Apple's open source mDNSResponder (https://opensource.apple.com/tarballs/mDNSResponder/).

* LICENSE:
  * license/LICENSE.dnsinfo.txt (Apple Public Source License 2.0)
  * HOMEPAGE:
  * Apache License
  * Version 2.0, January 2004
  * https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [{yyyy} {name of copyright owner}]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
(BSD License: https://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Webbit nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
 * Copyright (c) 2004-2007 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 * */
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Protocol Buffers - Google's data interchange format
Copyright 2013 Google Inc. All rights reserved.
https://developers.google.com/protocol-buffers/

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

This copy of Aalto XML processor is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

https://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/

Apache Harmony

Copyright 2006, 2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

1.122 crc32-stream 2.0.0 2.0.0

1.122.1 Available under license :

Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.123 binary-extensions 2.1.0 2.1.0
1.123.1 Available under license:

MIT License

Copyright (c) 2019 Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com), Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.124 mongodb-driver 3.12.4

1.124.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-jar/com/mongodb/client/internal/MongoChangeStreamCursorImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-jar/com/mongodb/ReplicaSetStatus.java
Open Source Used In DNAC Platform Arica 590

* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/BulkWriteUpsert.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/model/DBCollectionUpdateOptions.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/jndi/MongoClientFactory.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/gridfs/GridFSFindIterable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/MongoClientDelegate.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/Crypt.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/ReplaceRequest.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/model/DBCollectionCountOptions.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/AggregationOptions.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/DBDecoder.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/gridfs/GridFSDownloadStreamImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/GroupCommand.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/DistinctIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/gridfs/package-info.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/LazyDBDecoder.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/BulkWriteOperation.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/AcknowledgedBulkWriteResult.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/CollectionInfoRetriever.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/MappingIterable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/QueryBuilder.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/gridfs/GridFSInputFile.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/util/JSONSerializers.java
Open Source Used In DNAC Platform Articles 591

* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/ClientSessionBinding.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/LazyDBList.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/ListCollectionsIterable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/DBAddress.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/gridfs/GridFSDownloadStream.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/gridfs/GridFSFindIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/LazyDBEncoder.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/Crypts.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/KeyRetriever.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/CryptConnection.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/model/DBCollectionDistinctOptions.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/BulkWriteError.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/DBDecoderAdapter.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/AggregateIterable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/Cursor.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/DBCollectionObjectFactory.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/DBEncoderFactory.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/MongoClient.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/DBEncoderAdapter.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/BulkWriteException.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1.jar/com/mongodb/client/internal/SimpleMongoClient.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/hiertar/ListIndexesItetrableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/CommandMarker.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/MongoClient.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/FindIterltable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/MongoCollection.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/RemoveRequest.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/BulkWriteRequestBuilder.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/CryptBinding.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/model/DBCollectionRemoveOptions.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/ClientEncryptionImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/ChangeStreamIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/MongoDatabaseImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/vault/package-info.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/gridfs/CLI.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/MapReduceIterable.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/OperationExecutor.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/UnacknowledgedWriteResult.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/ChangeStreamIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/MongoDatabaseImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/vault/package-info.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/gridfs/CLI.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/ClientSession.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8MapReduceIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8AggregateIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8ListDatabasesIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/ClientSessionClock.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8ChangeStreamIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/package-info.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/MongoIterables.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8FindIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/MongoIterables.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8FindIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
jar/com/mongodb/client/internal/Java8ListCollectionsIterableImpl.java
* /opt/cola/permits/1136447742_1613767064.98/0/mongodb-driver-3-12-4-sources-1-
1.125 atob 2.1.2 2.1.2

1.125.1 Available under license:

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in
timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with
respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party
or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF
6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings,
agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark “Creative Commons” or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons’ then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at http://creativecommons.org/

At your option you may choose either of the following licenses:

* The MIT License (MIT)
* The Apache License 2.0 (Apache-2.0)
The MIT License (MIT)

Copyright (c) 2015 AJ ONeal

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 AJ ONeal

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.126 zlib 1.2.11

1.126.1 Available under license :

/* zlib.h -- interface of the 'zlib' general purpose compression library version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

*/
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.127 check-error 1.0.2

1.127.1 Available under license:

Copyright (c) 2013 Jake Luer <jake@aloggicalparadox.com> (http://aloggicalparadox.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights...
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.128 jopt-simple 5.0.4
1.128.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
The MIT License

Copyright (c) 2004-2015 Paul R. Holser, Jr.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1177945013_1625575335.76/0/jopt-simple-5-0-4-sources-8-
No license file was found, but licenses were detected in source scan.

/*
The MIT License

Copyright (c) 2004-2014 Paul R. Holser, Jr.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):
* /opt/cola/permits/1177945013_1625575335.76/0/jopt-simple-5-0-4-sources-8-jar/joptsimple/util/EnumConverter.java

1.129 hadoop-mapreduce-client-jobclient

3.2.2

1.129.1 Available under license :

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leventdb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4,
which has the following notices:

    The Netty Project
    =================

Please visit the Netty web site for more information:

* http://netty.io/
Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

-------------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:
This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
  * HOMEPAGE:
    * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
  * HOMEPAGE:
    * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:
This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:
This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging
framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4,
which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6,
which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17,
which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
  Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
  TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0,
which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ===============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.
* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* ============
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
============
Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
Later we found that vectorfree.com image is not properly licensed. The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0. The license is bundled with the download available here: http://www.vectorportal.com/subcategory/205/KILLER-WHALE-FREE-VECTOR.eps ifile/9136/detailtest.asp
--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0, which has the following notices:

 Apache Phoenix
 Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright: Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
The binary distribution of this product bundles binaries of Plexus Cipher: encryption/decryption Component 1.4, which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of software.amazon.ion:ion-java 1.0.1, which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of joda-time:joda-time:2.9.9 which has the following notices:
* =============================================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  =============================================================================

  This product includes software developed by
  Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of Ehcache 3.3.1, which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of snakeyaml (https://bitbucket.org/asomov/snakeyaml), which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of swagger-annotations (https://github.com/swagger-api/swagger-core), which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL
 * nor the names of its contributors may be used to endorse or
* promote products derived from this software without specific prior
    * written permission.

    * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
    * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
    * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
    * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
    * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
    * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
    * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
    * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
    * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
    * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
    * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
    * POSSIBILITY OF SUCH DAMAGE.

    */

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor
  the names of its contributors may be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++
 *
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 *
 * Abstract: The main routine
 *
 */

For src/main/native/src/org/apache/hadoop/io/compress/lz4/[lz4.h,lz4.c,lz4hc.h,lz4hc.c],

/*
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
      notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
      copyright notice, this list of conditions and the following disclaimer
      in the documentation and/or other materials provided with the
      distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at :
- LZ4 source repository : http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

For hadoop-common-project/hadoop-common/src/main/native/gtest

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
re2j 1.1

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following
license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

--------------------------------------------------------------------------------

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
- bootstrap v3.3.6
- broccoli-asset-rev v2.4.2
- broccoli-funnel v1.0.1
- datatables v1.10.8
- em-helpers v0.5.13
- em-table v0.1.6
- ember v2.2.0
- ember-array-contains-helper v1.0.2
- ember-bootstrap v0.5.1
- ember-cli v1.13.13
- ember-cli-app-version v1.0.0
- ember-cli-babel v5.1.6
- ember-cli-content-security-policy v0.4.0
- ember-cli-dependency-checker v1.2.0
- ember-cli-htmlbars v1.0.2
- ember-cli-htmlbars-inline-precompile v0.3.1
- ember-cli-ic-ajax v0.2.1
- ember-cli-inject-live-reload v1.4.0
- ember-cli-jquery-ui v0.0.20
- ember-cli-qunit v1.2.1
- ember-cli-release v0.2.8
- ember-cli-shims v0.0.6
- ember-cli-sri v1.2.1
- ember-cli-test-loader v0.2.1
- ember-cli-uglify v1.2.0
- ember-d3 v0.1.0
- ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js

---

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

---

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery
Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

For:
oj! Algorithms - version 43.0
--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as
documented below:

====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are
externally maintained libraries used by this software which have their
own licenses; we recommend you read them, as their terms may differ from
the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js
--------------------------------------------------------------------------------
Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

--------------------------------------------------------------------------------

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQldb Database 2.3.4

--------------------------------------------------------------------------------------------------------------------------

(HSQL License)
"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.
1.8. **Licensable** means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. **Modifications** means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from, or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. **Original Software** means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. **Patent Claims** means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. **Source Code** means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. **You** (or **Your** means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of
Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every
Contributor for any liability incurred by the Initial Developer or such
Contributor as a result of any such terms You offer.

3.6. Larger Works. You
may create a Larger Work by combining Covered Software with other code not
governed by the terms of this License and distribute the Larger Work as a single
product. In such a case, You must make sure the requirements of this License are
fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may
publish revised and/or new versions of this License from time to time. Each
version will be given a distinguishing version number. Except as provided in
Section 4.3, no one other than the license steward has the right to modify this
License.

4.2. Effect of New Versions. You may always continue to use,
distribute or otherwise make the Covered Software available under the terms of
the version of the License under which You originally received the Covered
Software. If the Initial Developer includes a notice in the Original Software
prohibiting it from being distributed or otherwise made available under any
subsequent version of the License, You must distribute and make the Covered
Software available under the terms of the version of the License under which You
originally received the Covered Software. Otherwise, You may also choose to use,
distribute or otherwise make the Covered Software available under the terms of
any subsequent version of the License published by the license
steward.

4.3. Modified Versions. When You are an Initial Developer and You
want to create a new license for Your Original Software, You may create and use
a modified version of this License if You: (a)rename the license and remove
any references to the name of the license steward (except to note that the
license differs from this License); and (b)otherwise make it clear that the
license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS,
WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT
LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS,
MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY
COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER
OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR
CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or
with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is
offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.
7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance
with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffers Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:

JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:
a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to
distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in
connection with its distribution of the Program in a commercial product
offering. The obligations in this section do not apply to any claims or Losses
relating to any actual or alleged intellectual property infringement. In order
to qualify, an Indemnified Contributor must: a) promptly notify the Commercial
Contributor in writing of such claim, and b) allow the Commercial Contributor to
control, and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may participate in
any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If that
Commercial Contributor then makes performance claims, or offers warranties
related to Product X, those performance claims and warranties are such
Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial
Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using and
distributing the Program and assumes all risks associated with its exercise of
rights under this Agreement, including but not limited to the risks and costs
of program errors, compliance with applicable laws, damage to or loss of data,
programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable
law, it shall not affect the validity or enforceability of the remainder of the
terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such
provision valid and enforceable.
If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveledbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

--------------------------------------------------------------------------------
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
"Licensor" means the individual or entity that offers the Work under the terms of this License.
"Original Author" means the individual or entity who created the Work.
"Work" means the copyrightable work of authorship offered under the terms of this License.
"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under
this License despite a previous violation.
2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or
phonorecord of the Work You distribute, publicly display, publicly perform, or
publicly digitally perform. You may not offer or impose any terms on the Work
that alter or restrict the terms of this License or the recipients' exercise of
the rights granted hereunder. You may not sublicense the Work. You must keep
intact all notices that refer to this License and to the disclaimer of
warranties. You may not distribute, publicly display, publicly perform, or
publicly digitally perform the Work with any technological measures that control
access or use of the Work in a manner inconsistent with the terms of this
License Agreement. The above applies to the Work as incorporated in a Collective
Work, but this does not require the Collective Work apart from the Work itself
to be made subject to the terms of this License. If You create a Collective
Work, upon notice from any Licensor You must, to the extent practicable, remove
from the Collective Work any credit as required by clause 4(b), as requested. If
You create a Derivative Work, upon notice from any Licensor You must, to the
extent practicable, remove from the Derivative Work any credit as required by
clause 4(b), as requested.
If you distribute, publicly display, publicly perform, or publicly digitally
perform the Work or any Derivative Works or Collective Works, You must keep
intact all copyright notices for the Work and provide, reasonable to the medium
or means You are utilizing: (i) the name of the Original Author (or pseudonym,
if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor
designate another party or parties (e.g. a sponsor institute, publishing entity,
journal) for attribution in Licensor's copyright notice, terms of service or by
other reasonable means, the name of such party or parties; the title of the Work
if supplied; to the extent reasonably practicable, the Uniform Resource
Identifier, if any, that Licensor specifies to be associated with the Work,
unless such URI does not refer to the copyright notice or licensing information
for the Work; and in the case of a Derivative Work, a credit identifying the use
of the Work in the Derivative Work (e.g., "French translation of the Work by
Original Author," or "Screenplay based on original Work by Original Author").
Such credit may be implemented in any reasonable manner; provided, however, that
in the case of a Derivative Work or Collective Work, at a minimum such credit
will appear where any other comparable authorship credit appears and in a manner
at least as prominent as such other comparable authorship credit.
5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS
THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND
CONCERNING
THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT
LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY,
OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME
JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH
EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN
NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL,
INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:
any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become
effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities
3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without
warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new
versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:

JDOM 1.1

/***************************************************************************/

/*--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

/***************************************************************************/
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.
The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
b. "Collection" means a collection of literary or artistic works, such as
encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the
exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party
or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URL if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.
6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE 
LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR 
ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES 
 ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS 
BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate 
automatically upon any breach by You of the terms of this License. 
Individuals or entities who have received Adaptations or Collections 
from You under this License, however, will not have their licenses 
terminated provided such individuals or entities remain in full 
compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will 
survive any termination of this License.
b. Subject to the above terms and conditions, the license granted here is 
perpetual (for the duration of the applicable copyright in the Work). 
Notwithstanding the above, Licensor reserves the right to release the 
Work under different license terms or to stop distributing the Work at 
any time; provided, however that any such election will not serve to 
withdraw this License (or any other license that has been, or is 
required to be, granted under the terms of this License), and this 
License will continue in full force and effect unless terminated as 
stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, 
the Licensor offers to the recipient a license to the Work on the same 
terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor 
offers to the recipient a license to the original Work on the same 
terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under 
applicable law, it shall not affect the validity or enforceability of 
the remainder of the terms of this License, and without further action 
by the parties to this agreement, such provision shall be reformed to 
the minimum extent necessary to make such provision valid and 
enforceable.
d. No term or provision of this License shall be deemed waived and no 
breach consented to unless such waiver or consent shall be in writing 
and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with 
respect to the Work licensed here. There are no understandings, 
agreements or representations with respect to the Work not specified 
here. Licensor shall not be bound by any additional provisions that 
may appear in any communication from You. This License may not be
modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this
License were drafted utilizing the terminology of the Berne Convention
for the Protection of Literary and Artistic Works (as amended on
September 28, 1979), the Rome Convention of 1961, the WIPO Copyright
Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996
and the Universal Copyright Convention (as revised on July 24, 1971).
These rights and subject matter take effect in the relevant
jurisdiction in which the License terms are sought to be enforced
according to the corresponding provisions of the implementation of
those treaty provisions in the applicable national law. If the
standard suite of rights granted under applicable copyright law
includes additional rights not granted under this License, such
additional rights are deemed to be included in the License; this
License is not intended to restrict the license of any rights under
applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty
whatsoever in connection with the Work. Creative Commons will not be
liable to You or any party on any legal theory for any damages
whatsoever, including without limitation any general, special,
incidental or consequential damages arising in connection to this
license. Notwithstanding the foregoing two (2) sentences, if Creative
Commons has expressly identified itself as the Licensor hereunder, it
shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the
Work is licensed under the CCPL, Creative Commons does not authorize
the use by either party of the trademark “Creative Commons” or any
related trademark or logo of Creative Commons without the prior
written consent of Creative Commons. Any permitted use will be in
compliance with Creative Commons' then-current trademark usage
guidelines, as may be published on its website or otherwise made
available upon request from time to time. For the avoidance of doubt,
this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

For: hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
/server/datanode/checker/AbstractFuture.java and
hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
/server/datanode/checker/TimeoutFuture.java

Copyright (C) 2007 The Guava Authors
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.130 is-descriptor 1.0.2
1.130.1 Available under license :
   The MIT License (MIT)

   Copyright (c) 2015-2017, Jon Schlinkert.

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.131 iceberg-common 0.8.0-incubating
1.131.1 Available under license :

   Apache Iceberg (incubating)
   Copyright 2017-2019 The Apache Software Foundation

   This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

------------------------------------------------------------------------------------------------------------------------

This project includes code from Kite, developed at Cloudera, Inc. with the following copyright notice:

<table>
<thead>
<tr>
<th>Copyright 2013 Cloudera Inc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensed under the Apache License, Version 2.0 (the &quot;License&quot;);</td>
</tr>
<tr>
<td>you may not use this file except in compliance with the License.</td>
</tr>
<tr>
<td>You may obtain a copy of the License at</td>
</tr>
<tr>
<td><a href="http://www.apache.org/licenses/LICENSE-2.0">http://www.apache.org/licenses/LICENSE-2.0</a></td>
</tr>
<tr>
<td>Unless required by applicable law or agreed to in writing, software</td>
</tr>
<tr>
<td>distributed under the License is distributed on an &quot;AS IS&quot; BASIS,</td>
</tr>
<tr>
<td>WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.</td>
</tr>
<tr>
<td>See the License for the specific language governing permissions and</td>
</tr>
<tr>
<td>limitations under the License.</td>
</tr>
</tbody>
</table>

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
you may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

------------------------------------------------------------------------------
This product includes a gradle wrapper.

* gradlew and gradle/wrapper/gradle-wrapper.properties

Copyright: 2010-2019 Gradle Authors.
Home page: https://github.com/gradle/gradle
License: http://www.apache.org/licenses/LICENSE-2.0

------------------------------------------------------------------------------
This product includes code from Apache Avro.

* Conversion in DecimalWriter is based on Avro's Conversions.DecimalConversion.

This product includes code from Apache Parquet.

* DynMethods.java
* DynConstructors.java
* AssertHelpers.java

Home page: https://parquet.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

This product includes code from Cloudera Kite.

* SchemaVisitor and visit methods

Copyright: 2013-2017 Cloudera Inc.
Home page: https://kitesdk.org/
License: http://www.apache.org/licenses/LICENSE-2.0

This product includes code from Presto.

* Retry wait and jitter logic in Tasks.java

Copyright: 2016 Facebook and contributors
Home page: https://prestodb.io/
License: http://www.apache.org/licenses/LICENSE-2.0

This product includes code from Apache iBATIS.

* Hive ScriptRunner.java

Copyright: 2004 Clinton Begin
Home page: http://ibatis.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

This product includes code from Apache Hive.
* Hive metastore derby schema in hive-schema-3.1.0.derby.sql

Copyright: 2011-2018 The Apache Software Foundation
Home page: http://hive.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

This product includes code from Apache Spark.

* dev/check-license script
* vectorized reading of definition levels in BaseVectorizedParquetValuesReader.java

Copyright: 2011-2018 The Apache Software Foundation
Home page: http://hive.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

1.132 winston-transport 3.0.1 3.0.1

1.132.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Charlie Robbins & the contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.133 text-table 0.2.0
1.133.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE.

1.134 get-intrinsic 1.1.1 1.1.1

1.134.1 Available under license:

MIT License

Copyright (c) 2020 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE.
1.135 xmlDom 0.2.1 0.2.1
1.135.1 Available under license:
You can choose any one of those:

The MIT License (MIT):
link:http://opensource.org/licenses/MIT

LGPL:
http://www.gnu.org/licenses/lgpl.html

1.136 fast-json-stable-stringify 2.1.0 2.1.0
1.136.1 Available under license:
This software is released under the MIT license:

Copyright (c) 2017 Evgeny Poberezkin
Copyright (c) 2013 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.137 winser-with-api 1.0.1 1.0.1
1.137.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.138 minimatch 3.0.4 3.0.4

1.138.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.139 proxy-addr 2.0.6 2.0.6

1.139.1 Available under license :

(The MIT License)

Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.140 fill-range 2.2.4

1.140.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.141 brace 0.11.1 0.11.1

1.141.1 Available under license:

Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.142 debug 4.3.2

1.142.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.143 babel-plugin-transform-es2015-for-of
6.23.0

1.143.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.144 okhttp 3.9.1

1.144.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and

   You must cause any modified files to carry prominent notices stating that You changed the files; and

   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.145 infinity-agent 1.0.2 1.0.2

1.145.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
Open Source Used In DNAC Platform Arica  705

1.146 readable-stream 2.3.6 2.3.6

1.146.1 Available under license:

Node.js is licensed for use as follows:

```text
***

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.
***
```

This license applies to parts of Node.js originating from the
https://github.com/joyent/node repository:

```text
***

Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
```
1.147 is-buffer 1.1.6 1.1.6

1.147.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.148 iconv-lite 0.4.23 0.4.23

1.148.1 Available under license:
Copyright (c) 2011 Alexander Shtuchkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.149 p-limit 2.3.0 2.3.0
1.149.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.150 axios 0.19.2 0.19.2
1.150.1 Available under license :
Copyright (c) 2014-present Matt Zabriskie

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.151 wtf-8 1.0.0 1.0.0

1.151.1 Available under license :

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.152 drill-format-syslog 1.19.0

1.152.1 Available under license :

Drill : Contrib : Format : Syslog

Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.153 content-disposition 0.5.1 0.5.1

1.153.1 Available under license:
(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.154 jsreport-reports 2.5.1 2.5.1

1.154.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs
whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version"
applies to it, you have the option of following the terms and
conditions either of that published version or of any later version
published by the Free Software Foundation. If the Library as you
received it does not specify a version number of the GNU Lesser
General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the
Library.

1.155 regenerate 1.4.1
1.155.1 Available under license:
MIT

1.156 puppeteer 1.11.0
1.156.1 Available under license:

    Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2017 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.157 drill-logical 1.19.0

1.157.1 Available under license:

Drill : Logical Plan
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.158 socket.io 1.7.4 1.7.4

1.158.1 Available under license :
(The MIT License)

Copyright (c) 2014-2016 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.159 fs-extra 1.0.0 1.0.0

1.159.1 Available under license :
(The MIT License)

Copyright (c) 2011-2016 JP Richardson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files
(the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.160 commons-cli 1.4

1.160.1 Available under license:
Apache Commons CLI
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.161 parseqs 0.0.5 0.0.5

1.161.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Gal Koren

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.162 batik-svgpp 1.13

1.162.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for Standardization for the definition of character entities used in the software’s documentation.

This product includes images from the Tango Desktop Project (http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme (http://www.jesusda.com/projects/pasodoble).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed subsequently incorporated within the Work.
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.163 lockfile 1.0.4 1.0.4
1.163.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.164 groovy-nio 3.0.8

1.164.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1218316510_1634924928.45/0/groovy-nio-3-0-8-sources-jar/org/apache/groovy/nio/runtime/WritablePath.java
* /opt/cola/permits/1218316510_1634924928.45/0/groovy-nio-3-0-8-sources-jar/org/apache/groovy/nio/extensions/NioExtensions.java
* /opt/cola/permits/1218316510_1634924928.45/0/groovy-nio-3-0-8-sources-
1.165 leveldbjni-all 1.8
1.165.1 Available under license:
Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0'

1.166 batch 0.4.0 0.4.0
1.166.1 Available under license:

Copyright (c) 2013 Nathan Rajlich &lt;nathan@tootallnate.net&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.167 bson 1.0.9 1.0.9
1.167.1 Available under license:
 Apache License
 Version 2.0, January 2004
 http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.168 isorelax 20090621
1.168.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: IslandSchemaReader.java,v 1.4 2001/11/01 00:11:45 kkawa Exp $
 * Copyright 2001 KAWAGUCHI Kohsuke
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
 * in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 * IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 * CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 * TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 * SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
 * /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-jar/org/iso_relax/dispatcher/IslandSchemaReader.java
No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: IgnoredSchema.java,v 1.5 2003/05/30 23:46:32 kkawa Exp $
 * Copyright 2001 Kohsuke KAWAGUCHI
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
* IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
* TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
* SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1.jar/org/iso_relax/dispatcher/impl/IgnoredSchema.java

No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: Dispatcher.java,v 1.6 2003/05/30 23:46:32 kkawa Exp $
 *
 * Copyright 2001 MURATA Makoto, KAWAGUCHI Kohsuke
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
 * in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 * IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 * CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 * TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 * SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):

Open Source Used In DNAC Platform Arica 744
No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: IslandVerifier.java,v 1.6 2003/05/30 23:46:32 kkawa Exp $
 *
 * Copyright 2001 MURATA Makoto, KAWAGUCHI Kohsuke
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
 * in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 * IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 * CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 * TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 * SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
/opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1.jar/org/iso_relax/dispatcher/IslandVerifier.java

No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: AttributesDecl.java,v 1.3 2001/11/01 00:11:45 kkawa Exp $
 *
 * Copyright 2001 Kohsuke KAWAGUCHI
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
* IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
* TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
* SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-
jar/org/iso_relax/dispatcher/AttributesDecl.java

No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: SchemaProvider.java,v 1.6 2003/05/30 23:46:32 kkawa Exp $
 *
 * Copyright 2001 Kohsuke KAWAGUCHI
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
 * in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 * IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 * CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 * TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 * SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-
jar/org/iso_relax/dispatcher/SchemaProvider.java

No license file was found, but licenses were detected in source scan.

/*
 * @(#)Id: ElementDecl.java,v 1.3 2001/11/01 00:11:45 kkawa Exp $
* Copyright 2001 Kohsuke KAWAGUCHI
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
* IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
* TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
* SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-
ar/jar/org/iso_relax/dispatcher/ElementDecl.java
No license file was found, but licenses were detected in source scan.

/*
* @(#)Id: IslandSchema.java,v 1.5 2003/05/30 23:46:32 kkawa Exp $
* *
* Copyright 2001 KAWAGUCHI Kohsuke
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
* IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
*
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
* TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
* SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
* /

Found in path(s):
* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-
  jar/org/iso_relax/dispatcher/IslandSchema.java
No license file was found, but licenses were detected in source scan.

/ *
* @(#)Id: DispatcherImpl.java,v 1.5 2003/05/30 23:46:32 kkawa Exp *
* *
* Copyright 2001 Kohsuke KAWAGUCHI
* *
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
* *
* The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.
* *
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
* IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
* TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
* SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
* /

Found in path(s):
* /opt/cola/permits/1135864262_1613618257.27/0/isorelax-20090621-sources-1-
  jar/org/iso_relax/dispatcher/impl/DispatcherImpl.java

1.169.objenesis 2.1
1.169.1 Available under license :

// ------------------------------------------------------------------
// NOTICE file corresponding to the section 4d of The Apache License,
// Version 2.0, in this case for Objenesis
// ------------------------------------------------------------------
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.170 bean-validation-api 1.1.0.Final

1.171 xmlhttprequest-ssl 1.5.5 1.5.5
1.171.1 Available under license :

Copyright (c) 2010 passive.ly LLC

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
1.172 has-binary2 1.0.3 1.0.3

1.172.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Kevin Roark

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.173 repeating 2.0.1 2.0.1

1.173.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.174 has-cors 1.1.0 1.1.0

1.174.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.175 regenerator-transform 0.14.5

1.175.1 Available under license:
MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.176 lombok 1.18.20

1.176.1 Available under license:
Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

==============================================================================
Licenses for included components:

org.ow2.asm:asm
org.ow2.asm:asm-analysis
org.ow2.asm:asm-commons
org.ow2.asm:asm-tree
org.ow2.asm:asm-util
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------------------------

rzwitserloot/com.zwitserloot.cmdreader

Copyright 2010 Reinier Zwitserloot.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

------------------------------------------------------------------------------

rzwitserloot/lombok.patcher
Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

------------------------------------------------------------------------------

1.177 parquet-encoding 1.12.0

1.177.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licenser" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This project includes code from Daniel Lemire's JavaFastPFOR project in this
binary artifact. The "Lemire" bit packing classes produced by parquet-generator
are derived from the JavaFastPFOR project.

Copyright: 2013 Daniel Lemire
Home page: http://lemire.me/en/
Project page: https://github.com/lemire/JavaFastPFOR
License: Apache License Version 2.0 http://www.apache.org/licenses/LICENSE-2.0

1.178 in flight 1.0.6 1.0.6
1.178.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.179 rx-lite 3.1.2
1.179.1 Available under license:
No license file was found, but licenses were detected in source scan.

/* Copyright (c) Microsoft Open Technologies, Inc. All rights reserved. See License.txt in the project root for license information. */

Found in path(s):
* /opt/cola/permits/1136088163_1613675018.74/0/rx-lite-3-1-2-2-tgz/package/rx.lite.min.js
No license file was found, but licenses were detected in source scan.

Copyright (c) Microsoft Open Technologies, Inc. All rights reserved.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1136088163_1613675018.74/0/rx-lite-3-1-2-2-tgz/package/readme.md
No license file was found, but licenses were detected in source scan.

// Copyright (c) Microsoft Open Technologies, Inc. All rights reserved. See License.txt in the project root for license information.

Found in path(s):
* /opt/cola/permits/1136088163_1613675018.74/0/rx-lite-3-1-2-2-tgz/package/rx.lite.js

1.180 jetty 9.4.41.v20210516
1.180.1 Available under license:
This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution ‘originates’
from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

   a) Subject to the terms of this Agreement, each Contributor hereby grants
       Recipient a non-exclusive, worldwide, royalty-free copyright license to
       reproduce, prepare derivative works of, publicly display, publicly
       perform, distribute and sublicense the Contribution of such Contributor,
       if any, and such derivative works, in source code and object code form.

   b) Subject to the terms of this Agreement, each Contributor hereby grants
       Recipient a non-exclusive, worldwide, royalty-free patent license under
       Licensed Patents to make, use, sell, offer to sell, import and otherwise
       transfer the Contribution of such Contributor, if any, in source code and
       object code form. This patent license shall apply to the combination of
       the Contribution and the Program if, at the time the Contribution is
       added by the Contributor, such addition of the Contribution causes such
       combination to be covered by the Licensed Patents. The patent license
       shall not apply to any other combinations which include the Contribution.
       No hardware per se is licensed hereunder.

   c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other
intellectual property rights of any other entity. Each Contributor
disclaims any liability to Recipient for claims brought by any other
entity based on infringement of intellectual property rights or
otherwise. As a condition to exercising the rights and licenses granted

hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

the Work and Derivative Works thereof.
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-------
Oracle
The following artifacts are CDDL + GPLv2 with classpath exception. 
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions. 
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.
org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-------
Assorted

The UnixCrypt.java code implements the one way cryptography used by
Unix systems for simple password protection. Copyright 1996 Aki Yoshida,
modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt
for non-commercial or commercial purposes and without fee is
granted provided that the copyright notice appears in all copies.

1.181 browserslist 4.18.1
1.181.1 Available under license :
The MIT License (MIT)

Copyright 2014 Andrey Sitnik <andrey@sitnik.ru> and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.182 private 0.1.8
1.182.1 Available under license :
Copyright (c) 2014 Ben Newman <bn@cs.stanford.edu>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.183 nth-check 1.0.2 1.0.2
1.183.1 Available under license :
Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.184 kafka-avro-serializer 6.1.1

1.184.1 Available under license :

Confluent Community License Agreement
Version 1.0

This Confluent Community License Agreement Version 1.0 (the Agreement) sets forth the terms on which Confluent, Inc. (Confluent) makes available certain software made available by Confluent under this Agreement (the Software). BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO SUCH TERMS AND CONDITIONS, YOU MUST NOT USE THE SOFTWARE. IF YOU ARE RECEIVING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY. Licensee means you, an individual, or the entity on whose behalf you are receiving the Software.

1. LICENSE GRANT AND CONDITIONS.

1.1 License. Subject to the terms and conditions of this Agreement, Confluent hereby grants to Licensee a non-exclusive, royalty-free, worldwide, non-transferable, non-sublicenseable license during the term of this Agreement to: (a) use the Software; (b) prepare modifications and derivative works of the Software; (c) distribute the Software (including without limitation in source code or object code form); and (d) reproduce copies of the Software (the License). Licensee is not granted the right to, and Licensee shall not, exercise the License for an Excluded Purpose. For purposes of this Agreement, Excluded Purpose means making available any software-as-a-service, platform-as-a-service, infrastructure-as-a-service or other similar online service that competes with Confluent products or services that provide the Software.
1.2 Conditions. In consideration of the License, Licensees distribution of the Software is subject to the following conditions:

(a) Licensee must cause any Software modified by Licensee to carry prominent notices stating that Licensee modified the Software.

(b) On each Software copy, Licensee shall reproduce and not remove or alter all Confluent or third party copyright or other proprietary notices contained in the Software, and Licensee must provide the notice below with each copy.

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at http://www.confluent.io/confluent-community-license. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

1.3 Licensee Modifications. Licensee may add its own copyright notices to modifications made by Licensee and may provide additional or different license terms and conditions for use, reproduction, or distribution of Licensees modifications. While redistributing the Software or modifications thereof, Licensee may choose to offer, for a fee or free of charge, support, warranty, indemnity, or other obligations. Licensee, and not Confluent, will be responsible for any such obligations.

1.4 No Sublicensing. The License does not include the right to sublicense the Software, however, each recipient to which Licensee provides the Software may exercise the Licenses so long as such recipient agrees to the terms and conditions of this Agreement.

2. TERM AND TERMINATION. This Agreement will continue unless and until earlier terminated as set forth herein. If Licensee breaches any of its conditions or obligations under this Agreement, this Agreement will terminate automatically and the License will terminate automatically and permanently.

3. INTELLECTUAL PROPERTY. As between the parties, Confluent will retain all right, title, and interest in the Software, and all intellectual property rights therein. Confluent hereby reserves all rights not expressly granted to Licensee in this Agreement. Confluent hereby reserves all rights in its trademarks and service marks, and no licenses therein are granted in this Agreement.

4. DISCLAIMER. CONFLUENT HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE.
5. LIMITATION OF LIABILITY. CONFLUENT WILL NOT BE LIABLE FOR ANY DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, LOST PROFITS OR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, INDIRECT, OR DIRECT DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THE FOREGOING SHALL APPLY TO THE EXTENT PERMITTED BY APPLICABLE LAW.

6. GENERAL.

   6.1 Governing Law. This Agreement will be governed by and interpreted in accordance with the laws of the state of California, without reference to its conflict of laws principles. If Licensee is located within the United States, all disputes arising out of this Agreement are subject to the exclusive jurisdiction of courts located in Santa Clara County, California, USA. If Licensee is located outside of the United States, any dispute, controversy or claim arising out of or relating to this Agreement will be referred to and finally determined by arbitration in accordance with the JAMS International Arbitration Rules. The tribunal will consist of one arbitrator. The place of arbitration will be Palo Alto, California. The language to be used in the arbitral proceedings will be English. Judgment upon the award rendered by the arbitrator may be entered in any court having jurisdiction thereof.

   6.2 Assignment. Licensee is not authorized to assign its rights under this Agreement to any third party. Confluent may freely assign its rights under this Agreement to any third party.

   6.3 Other. This Agreement is the entire agreement between the parties regarding the subject matter hereof. No amendment or modification of this Agreement will be valid or binding upon the parties unless made in writing and signed by the duly authorized representatives of both parties. In the event that any provision, including without limitation any condition, of this Agreement is held to be unenforceable, this Agreement and all licenses and rights granted hereunder will immediately terminate. Waiver by Confluent of a breach of any provision of this Agreement or the failure by Confluent to exercise any right hereunder will not be construed as a waiver of any subsequent breach of that right or as a waiver of any other right.

Apache Kafka
Copyright 2016 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

# Jackson JSON processor
Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.salaranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. Java ClassMate library was originally written by Tatu Saloranta (tatu.salaranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="Home page of The Apache Software Foundation">
<link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
<link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png"/>
<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See
the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF
licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file
except in compliance with the License. You may obtain a copy of the License at.
http://www.apache.org/licenses/LICENSE-2.0. Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions and limitations under the License. -->

</head>

<body>

<!-- Navigation -->

<header>

<nav class="navbar navbar-default navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a href="#" class="navbar-brand"><span class="glyphicon glyphicon-home"></span></a>

</div>

<div class="collapse navbar-collapse" id="mainnav-collapse">

</div>

</div>

</nav>

</header>

<div class="container">

</div>

</body>
Apache License
Version 2.0, January 2004

<a href="http://www.apache.org/licenses/">http://www.apache.org/licenses/</a>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly
display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.</p>
<p><strong>3. Grant of Patent License</strong>. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.</p>
<p><strong>4. Redistribution</strong>. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:</p>
<ol style="list-style: lower-latin;">
<li>You must give any other recipients of the Work or Derivative Works a copy of this License; and</li>
<li>You must cause any modified files to carry prominent notices stating that You changed the files; and</li>
<li>You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and</li>
<li>If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.</li>
</ol>
provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor
harmless for any liability incurred by, or claims asserted against, such
Contributor by reason of your accepting any such warranty or additional
liability.</p>
</p>END OF TERMS AND CONDITIONS</p>

APPENDIX: How to apply the Apache License to your work<br>
To apply the Apache License to your work, attach the following boilerplate
notice, with the fields enclosed by brackets "[ ]" replaced with your own
identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also
recommend that a file or class name and description of purpose be included
on the same "printed page" as the copyright notice for easier
identification within third-party archives.</p>

Licensed under the Apache License, Version 2.0 (the License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an AS IS BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an AS IS BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an AS IS BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright [yyyy] [name of copyright owner]
This copy of Jackson JSON processor annotations is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.
You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

License: Apache-2

On Debian systems, the Apache 2.0 license can be found in /usr/share/common-licenses/Apache-2.0.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

This distribution has a binary dependency on jersey, which is available under the CDDL
License as described below.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL - Version 1.1)
1. Definitions.
1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor 
(if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files 
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by 
the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial 
grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing 
Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally 
released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this
License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of
this License are fulfilled for the Covered Software.

4. Versions of the License.
4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time.
Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the
license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the
version of the License under which You originally received the Covered Software. If the Initial Developer includes a
notice in the Original Software prohibiting it from being distributed or otherwise made available under any
subsequent version of the License, You must distribute and make the Covered Software available under the terms of
the version of the License under which You originally received the Covered Software. Otherwise, You may also
choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent
version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may
create and use a modified version of this License if You: (a) rename the license and remove any references to the
name of the license steward (except to note that the license differs from this License); and (b) otherwise make it
clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT
WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A
PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND
PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER
CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION.
THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE
OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.
6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms
herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their
nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R.  252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software
(except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Apache Avro
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
The following commands were used to generate license and notice files. Replace <VERSION> with the Schema Registry version, <SRC_PATH> with the path to the Schema Registry source directory, and <LICENSE_TOOL_PATH> with the path of the license tool.

cd <SRC_PATH>
mvn package -DskipTests
mkdir /tmp/jars
mkdir /tmp/overrides
cd <LICENSE_TOOL_PATH>
./bin/run_license_job.bash -i /tmp/jars -l <SRC_PATH>/licenses -n <SRC_PATH>/notices -h <SRC_PATH>/licenses-and-notices.html -o /tmp/overrides
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="en-US">
<h3>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0</h3>

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.
<li>
<p>
<b>1.5. &ldquo;Initial Developer&rdquo;</b> means the individual or entity that first makes Original Software available under this License.
</p>
</li>

<li>
<p>
<b>1.6. &ldquo;Larger Work&rdquo;</b> means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
</p>
</li>

<li>
<p>
<b>1.7. &ldquo;License&rdquo;</b> means this document.
</p>
</li>

<li>
<p>
<b>1.8. &ldquo;Licensable&rdquo;</b> means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
</p>
</li>

<li>
<p>
<b>1.9. &ldquo;Modifications&rdquo;</b> means the Source Code and Executable form of any of the following:
</p>
<ul>
<li>
<p>
<b>A.</b> Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
</p>
</li>
</ul>
</li>
B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. &ldquo;Original Software&rdquo; means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. &ldquo;Patent Claims&rdquo; means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. &ldquo;Source Code&rdquo; means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. &ldquo;You&rdquo; (or &ldquo;Your&rdquo;) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, includes any entity which controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

<li></li>
</ul></li>

<li><p><b>2. License Grants.</b></p></li>
<ul><li><p><b>2.1. The Initial Developer Grant.</b></p></li>

<p>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
</p><ul><li><p>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</p></li></ul>
<li><p>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
</p></li>
<li><p>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
</p></li>
</ul>
<b>(d)</b> Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.


2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(b)(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b)(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
<li><p>(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.</p></li>

<li><p>(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.</p></li>

<p><b>3. Distribution Obligations.</b></p>

<ul>
  <li><p><b>3.1. Availability of Source Code.</b></p>
  <p>Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.</p></li>
</ul>
<li><p><b>3.2. Modifications.</b></p></li>
<p>The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.</p>
</li>
<li><p><b>3.3. Required Notices.</b></p></li>
<p>You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.</p>
</li>
<li><p><b>3.4. Application of Additional Terms.</b></p></li>
<p>You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.</p>
</li>
<li><p><b>3.5. Distribution of Executable Versions.</b></p></li>
<p>You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received
the Covered Software. If the Initial Developer includes a
notice in the Original Software prohibiting it from being
distributed or otherwise made available under any
subsequent version of the License, You must distribute and
make the Covered Software available under the terms of the
version of the License under which You originally received
the Covered Software. Otherwise, You may also choose to
use, distribute or otherwise make the Covered Software
available under the terms of any subsequent version of the
License published by the license steward.
</p>
</li>
<li><p><b>4.3. Modified Versions.</b></p>
<p>When You are an Initial Developer and You want to
create a new license for Your Original Software, You may
create and use a modified version of this License if You:
(a)&nbsp;rename the license and remove any references to
the name of the license steward (except to note that the
license differs from this License); and (b)&nbsp;otherwise
make it clear that the license contains terms which differ
from this License.
</p></li>
<li><p><b>5. DISCLAIMER OF WARRANTY.</b></p>
<p>Covered Software is provided under this License on an
"as is" basis, without warranty of any kind, either
expressed or implied, including, without limitation,
wa...
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. § 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. § 12.212 (Sept. 1995). Consistent with 48 C.F.R. § 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides...
otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
<h3>The GNU General Public License (GPL) Version 2, June 1991</h3>

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable
copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
</p>
</li>
<li>
<p>c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
</p>
</li>
<ul>
<li>The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
</li>
<li>If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
</li>
</ul>
<li>
<p><b>4.</b> You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
</p>
</li>
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software
distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
</p>
<p>This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
</p>
</li>

<li>
<p><b>8.</b> If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
</p>
</li>

<li>
<p><b>9.</b> The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
</p>
</li>

<li>
<p>Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
</p>
</li>

<li>
<p><b>10.</b> If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

</p></li>

<li>
<p>NO WARRANTY</p>
</li>

<li>
<p>BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
</p></li>

<li>
<p>IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
</p></li>

<li>
<p>END OF TERMS AND CONDITIONS</p>
</li>

<li>
<p>How to Apply These Terms to Your New Programs</p>
</li>

<p>If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
</p>
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words
"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."
</p></div>

<p>Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.
</p>

<p>As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
</p>
Contributor.</blockquote>

1. Covered Software; means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.</blockquote>

1. Executable; means the Covered Software in any form other than Source Code.</blockquote>

1. Initial Developer; means the individual or entity that first makes Original Software available under this License.</blockquote>

1. Larger Work; means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.</blockquote>

1. License; means this document.</blockquote>

1. Licensable; means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.</blockquote>

1. Modifications; means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.</blockquote>

1. Original Software; means the Source Code and Executable form of computer software code that is originally released under this License.</blockquote>

1. Patent Claims; means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.</blockquote>

1. Source Code; means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</blockquote>

1. You; (or Your;) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</blockquote>

2. License Grants.</blockquote>

2.1. Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor to the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

<blockquote>3.5. Distribution of Executable Versions.</blockquote>

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

<blockquote>3.6. Larger Works.</blockquote>

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

<blockquote>4. Versions of the License.</blockquote>

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

<blockquote>5. DISCLAIMER OF WARRANTY.</blockquote>

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

<blockquote>6. TERMINATION.</blockquote>

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply
with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as &ldquo;Participant;&rdquo;) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a &ldquo;commercial item,&rdquo; as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of &ldquo;commercial computer software&quot; (as that term is defined at 48 C.F.R. &sect; 252.227-7014(a)(1)) and &ldquo;commercial computer software documentation;&rdquo; as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances. It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

<p>END OF TERMS AND CONDITIONS</p>

<p>How to Apply These Terms to Your New Programs</p>

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the &quot;copyright&quot; line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) &lt;year&gt; &lt;name of author&gt;

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a &quot;copyright disclaimer&quot; for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

&lt;li style="background-color:yellow;"&gt;

&lt;p&gt;&lt;strong&gt;&quot;CLASSPATH&quot;&lt;/strong&gt; EXCEPTION TO THE GPL VERSION 2&lt;/strong&gt;&lt;br /&gt;

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words &quot;Oracle designates this particular file as subject to the &quot;CLASSPATH&quot; exception as provided by
This copy of Jackson JSON processor is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"
The project is licensed under the Confluent Community License, except for client libs, which is under the Apache 2.0 license.

See LICENSE file in each subfolder for detailed license agreement.

Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. -->

<!-- Navigation -->
<header>

</header>
<body>

</body>
The Apache Software Foundation uses various licenses to distribute software and documentation, to accept regular contributions from individuals and corporations, and to accept larger grants of existing software products. These licenses help us achieve our goal of providing reliable and long-lived software products through collaborative open source software development. In all cases, contributors retain full rights to use their original contributions for any other purpose outside of Apache while providing the ASF and its projects the right to distribute and build upon their work within Apache.

Licensing of Distributions

All software produced by The Apache Software Foundation or any of its
Open Source Used In DNAC Platform Articles

projects or subjects is licensed according to the terms of the documents listed below.</p>

<h3 id="2.0">Apache License, Version 2.0 (current)<a class="headerlink" href="#2.0" title="Permanent link">&para;</a></h3>

<p><a href="LICENSE-2.0">http://www.apache.org/licenses/LICENSE-2.0</a> (TXT or HTML)</p>

<p>The 2.0 version of the Apache License was approved by the ASF in 2004. The goals of this license revision have been to reduce the number of frequently asked questions, to allow the license to be reusable without modification by any project (including non-ASF projects), to allow the license to be included by reference instead of listed in every file, to clarify the license on submission of contributions, to require a patent license on contributions that necessarily infringe the contributor’s own patents, and to move comments regarding Apache and other inherited attribution notices to a location outside the license terms (the NOTICE file).</p>

<p>The result is a license that is supposed to be compatible with other open source licenses, while remaining true to the original goals of the Apache Group and supportive of collaborative development across both nonprofit and commercial organizations. The Apache Software Foundation is still trying to determine if this version of the Apache License is compatible with the GPL.</p>

<p>All packages produced by the ASF are implicitly licensed under the Apache License, Version 2.0, unless otherwise explicitly stated. More developer documentation on how to apply the Apache License to your work can be found in *Applying the Apache License, Version 2.0*</p>

<h3 id="1.1">Apache License, Version 1.1 (historic)<a class="headerlink" href="#1.1" title="Permanent link">&para;</a></h3>

<p><a href="LICENSE-1.1">http://www.apache.org/licenses/LICENSE-1.1</a></p>

<p>The 1.1 version of the Apache License was approved by the ASF in 2000. The primary change from the 1.0 license is in the ‘advertising clause’ (section 3 of the 1.0 license); derived products are no longer required to include attribution in their advertising materials, only in their documentation.</p>

<p>Individual packages licensed under the 1.1 version may have used different wording due to varying requirements for attribution or mark identification, but the binding terms were all the same.</p>

<h3 id="1.0">Apache License, Version 1.0 (historic)<a class="headerlink" href="#1.0" title="Permanent link">&para;</a></h3>

<p><a href="LICENSE-1.0">http://www.apache.org/licenses/LICENSE-1.0</a></p>

<p>This is the original Apache License which applies only to older versions of Apache packages (such as version 1.2 of the Web server).</p>

<h1 id="clas">Contributor License Agreements<a class="headerlink" href="#clas" title="Permanent link">&para;</a></h1>

<p>The ASF desires that all contributors of ideas, code, or documentation to any Apache projects complete, sign, and submit (via fax or email) an Individual Contributor License Agreement (ICLA). The purpose of this agreement is to clearly define the
terms under which intellectual property has been contributed to the ASF and thereby allow us to defend the project should there be a legal dispute regarding the software at some future time. A signed ICLA is required to be on file before an individual is given commit rights to an ASF project.</p><p>For a corporation that has assigned employees to work on a Apache project, a Corporate CLA (CCLA) is available for contributing intellectual property via the corporation, that may have been assigned as part of an employment agreement. Note that a Corporate CLA does not remove the need for every developer to sign their own ICLA as an individual, to cover any of their contributions which are not owned by the corporation signing the CCLA.</p><p>The ICLA is not tied to any employer you may have, so it is recommended to use one's personal email address in the contact details, rather than an @work address. </p><p>Your Full name will be published unless you provide an alternative Public name. For example if your full name is Andrew Bernard Charles Dickens, but you wish to be known as Andrew Dickens, please enter the latter as your Public name.</p><p>The email address and other contact details are not published.</p><p>When an individual or corporation decides to donate a body of existing software or documentation to one of the Apache projects, they need to execute a formal Software Grant Agreement (SGA) with the ASF. Typically, this is done after negotiating approval with the ASF Incubator or one of the PMCs, since the ASF will not accept software unless there is a viable community available to support a collaborative project.</p><p>Documents may be submitted by fax or email. If submitting by fax, please print, sign, and fax all pages of the document to +1-919-573-9199. Please send documents right-side-up, first page first; and send only one document per fax.</p><p>If submitting by email, please fill the form with a pdf viewer, print, sign, scan all pages into a single pdf file, and email the pdf file as an attachment to secretary@apache.org. If possible, send the attachment from the email address in the document. Please send only one document per email.</p><p>If you prefer to sign electronically, please fill the form, save it locally (e.g. icla.pdf), and sign the file by preparing a detached PGP signature. For example,</p><p>gpg --armor --detach-sign icla.pdf</p><p>The above will create a file icla.pdf.asc. Send both the file and signature as attachments in the same email to secretary@apache.org. Please send only one document (file plus signature) per email. Please do not submit your public key to Apache. Instead, please upload your public key to pgpkeys.mit.edu.</p><p>The files typically are named icla.pdf and icla.pdf.asc for individual agreements;
ccla.pdf and ccla.pdf.asc for corporate agreements;
software-grant.pdf and software-grant.pdf.asc for grants. </p>

<h1 id="crypto">Export restrictions</h1>
For export restriction information, please consult our <a href="/licenses/exports/">ASF Export
Classifications</a> page. </p>

<h1 id="trademarks">Trademark and Logo Usage</h1>
For ASF trademark and logo usage information, please consult our <a href="/foundation/marks/">ASF
Trademark Use Policy</a> page. </p>

<h1 id="questions">Questions?</h1>
For answers to frequently asked licensing questions, please consult our
<a href="/foundation/license-faq.html">Licensing Frequently Asked Questions</a> page. </p>

</div>

<footer class="bg-primary">
<div class="container">
<div class="row">
<div class="col-sm-1"></div>

<div class="col-sm-2">
<h5 class="white">Community</h5>
<ul class="list-unstyled white" role="menu">
<li><a href="http://community.apache.org/">Overview</a></li>
<li><a href="/foundation/conferences.html">Conferences</a></li>
<li><a href="http://community.apache.org/newcomers/">Getting Started</a></li>
<li><a href="/foundation/how-it-works.html">The Apache Way</a></li>
<li><a href="/travel/">Travel Assistance</a></li>
<li><a href="/foundation/getinvolved.html">Get Involved</a></li>
<li><a href="http://community.apache.org/newbiefaq.html">Community FAQ</a></li>
</ul>
</div>

<div class="col-sm-2">
<h5 class="white">Innovation</h5>
<ul class="list-unstyled white" role="menu">
<li><a href="http://incubator.apache.org/">Incubator</a></li>
<li><a href="http://labs.apache.org/">Labs</a></li>
<li><a href="/licenses/">Licensing</a></li>
<li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
<li><a href="/foundation/marks/">Trademark Policy</a></li>
<li><a href="/foundation/contact.html">Contacts</a></li>
</ul>
</div>
This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.

For additional credits (generally to people who reported problems) see CREDITS file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.185 jws 3.1.5 3.1.5

1.185.1 Available under license :

Copyright (c) 2013 Brian J. Brennan

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the
Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.186 decamelize 2.0.0 2.0.0
1.186.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.187 domelementtype 1.3.1 1.3.1
1.187.1 Available under license:

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
1.188 body-parser 1.19.0 1.19.0

1.188.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.189 drill-storage-elasticsearch 1.19.0

1.189.1 Available under license:

Drill : Contrib : Storage : ElasticSearch
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.190 redux-saga 0.16.0 0.16.0

1.190.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Yassine Elouafi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.191 boolbase 1.0.0 1.0.0

1.191.1 Available under license :

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND ISC DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

1.192 jakarta-annotation-api 1.3.5

1.192.1 Available under license :

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.
"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual
property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.
3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this
Agreement and does not cure such failure in a reasonable period of
time after becoming aware of such noncompliance. If all Recipient's
rights under this Agreement terminate, Recipient agrees to cease use
and distribution of the Program as soon as reasonably practicable.
However, Recipient's obligations under this Agreement and any licenses
granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement,
but in order to avoid inconsistency the Agreement is copyrighted and
may only be modified in the following manner. The Agreement Steward
reserves the right to publish new versions (including revisions) of
this Agreement from time to time. No one other than the Agreement
Steward has the right to modify this Agreement. The Eclipse Foundation
is the initial Agreement Steward. The Eclipse Foundation may assign the
responsibility to serve as the Agreement Steward to a suitable separate
entity. Each new version of the Agreement will be given a distinguishing
version number. The Program (including Contributions) may always be
Distributed subject to the version of the Agreement under which it was
received. In addition, after a new version of the Agreement is published,
Contributor may elect to Distribute the Program (including its
Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient
receives no rights or licenses to the intellectual property of any
Contributor under this Agreement, whether expressly, by implication,
estoppel or otherwise. All rights in the Program not expressly granted
under this Agreement are reserved. Nothing in this Agreement is intended
to be enforceable by any entity that is not a Contributor or Recipient.
No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following
Secondary Licenses when the conditions for such availability set forth
in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),
version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A
is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to
look for such a notice.

You may add additional accurate notices of copyright ownership.

---
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original

Open Source Used In DNAC Platform Arica 867
authors’ reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients’ exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.
One line to give the program’s name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Jakarta Annotations

This content is produced and maintained by the Jakarta Annotations project.

* Project home: https://projects.eclipse.org/projects/ee4j.ca

## Trademarks

Jakarta Annotations is a trademark of the Eclipse Foundation.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:
* https://github.com/eclipse-ee4j/common-annotations-api

## Third-party Content

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.193 netty-resolver 4.1.59.Final

#### 1.193.1 Available under license :

No license file was found, but licenses were detected in source scan.

```java
/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

Found in path(s):

* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-jar/io/netty/resolver/HostsFileParser.java
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-jar/io/netty/resolver/DefaultAddressResolverGroup.java
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-jar/io/netty/resolver/DefaultHostsFileEntriesResolver.java
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-jar/io/netty/resolver/AddressResolver.java
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-jar/io/netty/resolver/DefaultNameResolver.java
```
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4.1.59-final-sources-jar/io/netty/resolver/HostsFileEntries.java
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4.1.59-final-sources-jar/io/netty/resolver/ResolvedAddressTypes.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449559_1616156337.74/0/netty-resolver-4-1-59-final-sources-
  jar/io/netty/resolver/RoundRobinInetAddressResolver.java

1.194 babel-helper-function-name 6.24.1
1.194.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.195 apib-include-directive 0.1.0 0.1.0

1.195.1 Available under license:
MIT License

Copyright (c) 2016 Daniel G. Taylor
Copyright (c) 2017 Keisuke Minami

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.196 raw-body 2.3.3 2.3.3

1.196.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013-2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.197 semver 5.7.1 5.7.1

1.197.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.198 json-glib 1.1.2-0ubuntu1

1.198.1 Available under license :
GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.
1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.
However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.
You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>AppendixA.License: JSON-GLib Reference Manual</title>
<meta name="generator" content="DocBook XSL Stylesheets V1.78.1">
<link rel="home" href="index.html" title="JSON-GLib Reference Manual">
<link rel="up" href="index.html" title="JSON-GLib Reference Manual">
<link rel="prev" href="annotation-glossary.html" title="Annotation Glossary">
<meta name="generator" content="GTK-Doc V1.24 (XML mode)">
<link rel="stylesheet" href="style.css" type="text/css">
</head>
<body bgcolor="white" text="black" link="#0000FF" vlink="#840084" alink="#0000FF">
<table class="navigation" id="top" width="100%" summary="Navigation header" cellpadding="2" cellspacing="5"><tr valign="middle">
<td width="100%" align="left" class="shortcuts"></td>
<td><a accesskey="h" href="index.html"><img src="home.png" width="16" height="16" border="0" alt="Home"></a></td>
<td><img src="up-insensitive.png" width="16" height="16" border="0"></td>
<td><a accesskey="p" href="annotation-glossary.html"><img src="left.png" width="16" height="16" border="0" alt="Prev"></a></td>
<td><img src="right-insensitive.png" width="16" height="16" border="0"></td>
</tr></table>

<p>This library is free software; you can redistribute it and/or modify it under the terms of the <em class="citetitle">GNU Library General Public License</em> as published by the Free Software Foundation.

---

Open Source Used In DNAC Platform Arica 893
Foundation; either version 2 of the License, or (at your option) 
any later version.

This library is distributed in the hope that it will be useful, 
but WITHOUT ANY WARRANTY; without even the implied warranty of 
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the 
GNU Library General Public License for more details.

You may obtain a copy of the GNU Library General 
Public License from the Free Software Foundation by 
visiting their Web site or by writing to:

FreeSoftwareFoundation,Inc.
59TemplePlace-Suite330
Boston,MA
02111-1307
USA

1.199 jetty-http 9.4.34.v20201102

1.199.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at 
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.

No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner thatreasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed...
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==============================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message
The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.200 @babel/parser 7.16.4
1.200.1 Available under license : 
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.201 bytes 0.2.1 0.2.1
1.201.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.202 hive-hbase-handler 3.1.2

1.202.1 Available under license:

Hive HBase Handler
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]"
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.203 strip-ansi 3.0.1 3.0.1

1.203.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.204 invariant 2.2.4 2.2.4

1.204.1 Available under license:
MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.205 async-replace 1.0.1 1.0.1

1.205.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.206 strip-ansi 0.1.1 0.1.1

1.206.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.207 jsonify 0.0.0

1.207.1 Available under license:

No license file was found, but licenses were detected in source scan.

# jsonify

This module provides Douglas Crockford's JSON implementation without modifying any globals.

`stringify` and `parse` are merely exported without respect to whether or not a global `JSON` object exists.

[![build status](https://secure.travis-ci.org/substack/jsonify.png)](http://travis-ci.org/substack/jsonify)

# methods
``` js
var json = require('jsonify');

## json.parse(source, reviver)

Return a new javascript object from a parse of the `source` string.

If a `reviver` function is specified, walk the structure passing each name/value pair to `reviver.call(parent, key, value)` to transform the `value` before parsing it.

## json.stringify(value, replacer, space)

Return a string representation for `value`.

If `replacer` is specified, walk the structure passing each name/value pair to `replacer.call(parent, key, value)` to transform the `value` before stringifying it.

If `space` is a number, indent the result by that many spaces.
If `space` is a string, use `space` as the indentation.

# install

With [npm](http://npmjs.org) do:

```
npm install jsonify
```

To use this module in the browser, check out [browserify](https://github.com/substack/node-browserify).

# license

public domain

Found in path(s):
* /opt/cola/permits/1108587829_1609900447.26/0/jsonify-master-zip/jsonify-master/readme.markdown

1.208.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.209 form-data 2.3.2 2.3.2
1.209.1 Available under license:
Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.210 apache-log4j 2.17.1

1.210.1 Available under license:

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.211 clone-response 1.0.2

1.211.1 Available under license :
MIT License

Copyright (c) 2017 Luke Childs

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.212 hbase-server 2.4.2

1.212.1 Available under license:

Apache HBase - Server
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

JUnit License

Common Public License Version 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.
"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

   a) it complies with the terms and conditions of this Agreement; and

   b) its license agreement:

      i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

      ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of
program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2017 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in http://www.unicode.org/copyright.html

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY
SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

All trademarks and registered trademarks mentioned herein are the
property of their respective owners.


# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.

# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
# 1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.
   2. Redistributions in binary form must reproduce the above
      copyright notice, this list of conditions and the following
      disclaimer in the documentation and/or other materials provided with
      the distribution.
   3. Neither the name of Google Inc. nor the names of its
      contributors may be used to endorse or promote products derived from
      this software without specific prior written permission.

# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
# LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
# CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
# SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
# BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
# LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
# SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# The word list in cjdict.txt are generated by combining three word lists
# listed below with further processing for compound word breaking. The
# frequency is generated with an iterative training against Google web
# corpora.

# * Libtabe (Chinese)
#   - https://sourceforge.net/project/?group_id=1519
#   - Its license terms and conditions are shown below.

# * IPADIC (Japanese)
#   - http://chasen.aist-nara.ac.jp/chasen/distribution.html
#   - Its license terms and conditions are shown below.

# ---------COPYING.libtabe ---- BEGIN-------------------
#/*
# Copyright (c) 1999 TaBE Project.
# Copyright (c) 1999 Pai-Hsiang Hsiao.

# * All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * Neither the name of the TaBE Project nor the names of its
# * contributors may be used to endorse or promote products derived
# * from this software without specific prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# /*
# * Copyright (c) 1999 Computer Systems and Communication Lab,
# * Institute of Information Science, Academia
# * Sinica. All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * Neither the name of the Computer Systems and Communication Lab
# * nor the names of its contributors may be used to endorse or
# * promote products derived from this software without specific
# * prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */

# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
#
# ---------------COPYING.libtable-----END-----------------------------
#
# ---------------COPYING.ipadic-----BEGIN-------------------------------
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

# NO WARRANTY

# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
 implied, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

# Each user of the program will agree and understand, and be deemed to
# have agreed and understood, that there is no warranty whatsoever for
# the program and, accordingly, the entire risk arising from or
# otherwise connected with the program is assumed by the user.

# Therefore, neither ICOT, the copyright holder, or any other
# organization that participated in or was otherwise related to the
# development of the program and their respective officials, directors,
# officers and other employees shall be held liable for any and all
# damages, including, without limitation, general, special, incidental
# and consequential damages, arising out of or otherwise in connection
# with the use or inability to use the program or any product, material
# or result produced or otherwise obtained by using the program,
# regardless of whether they have been advised of, or otherwise had
# knowledge of, the possibility of such damages at any time during the
# project or thereafter. Each user will be deemed to have agreed to the
# foregoing by his or her commencement of use of the program. The term
# "use" as used herein includes, but is not limited to, the use,
# modification, copying and distribution of the program and the
# production of secondary products from the program.

# In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.


3. Lao Word Break Dictionary Data (laodict.txt)

# Copyright (c) 2013 International Business Machines Corporation
# and others. All Rights Reserved.
# # Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
# (copied below)
#
# This file is derived from the above dictionary, with slight
# modifications.
# # ----------------------------------------------------------------------
# # Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# # All rights reserved.
# # # Redistribution and use in source and binary forms, with or without
# # modification,
# # are permitted provided that the following conditions are met:
# # # # Redistributions of source code must retain the above copyright notice, this
# # list of conditions and the following disclaimer. Redistributions in
# # binary form must reproduce the above copyright notice, this list of
# # conditions and the following disclaimer in the documentation and/or
# # other materials provided with the distribution.
# # # # THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# # "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# # LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# # FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# # COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# # INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# # (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# # SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# # HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# # STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# # ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# # OF THE POSSIBILITY OF SUCH DAMAGE.
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

# 7. Database Ownership

# This list is part of a project hosted at:
# github.com/kanyawtech/myanmar-karen-word-lists
# #
# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
# disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided
# with the distribution.
#
# Neither the name Myanmar Karen Word Lists, nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
# BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
# EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
# TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
# DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
# ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
# TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
# THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
# SUCH DAMAGE.
#
public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do
not apply to the TZ Database or contributions that individuals make
to it. Should any claims be made and substantiated against the TZ
Database, the organization that is providing the IANA
Considerations defined in this RFC, under the memorandum of
understanding with the IETF, currently ICANN, may act in accordance
with all competent court orders. No ownership claims will be made
by ICANN or the IETF Trust on the database or the code. Any person
making a contribution to the database or code waives all rights to
future claims in that contribution or in the TZ Database.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Compress
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The files in the package org.apache.commons.compress.archivers.sevenz were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/), which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)

JBoss (http://www.jboss.org) is distributed under the LGPL. Mockrunner includes the jboss-j2ee.jar and jboss-jaas.jar.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy,
and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
This product includes software developed by:
  o The Apache Software Foundation (http://www.apache.org/)

A. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING JYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Jython") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Jython alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2007 Python Software Foundation; All Rights Reserved" are retained in Jython alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Jython or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Jython.

4. PSF is making Jython available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF JYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF JYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING JYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Jython, Licensee agrees to be bound by the terms and conditions of this License Agreement.

Jython 2.0, 2.1 License
================================
Copyright (c) 2000-2009 Jython Developers.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Jython Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
JPython 1.1.x Software License.

IMPORTANT: PLEASE READ THE FOLLOWING AGREEMENT CAREFULLY.

BY CLICKING ON THE "ACCEPT" BUTTON WHERE INDICATED, OR BY INSTALLING,
COPYING OR OTHERWISE USING THE SOFTWARE, YOU ARE DEEMED TO HAVE AGREED TO
THE TERMS AND CONDITIONS OF THIS AGREEMENT.

JPython version 1.1.x

1. This LICENSE AGREEMENT is between the Corporation for National Research
Initiatives, having an office at 1895 Preston White Drive, Reston, VA
20191 ("CNRI"), and the Individual or Organization ("Licensee")
accessing and using JPython version 1.1.x in source or binary form and
its associated documentation as provided herein ("Software").

2. Subject to the terms and conditions of this License Agreement, CNRI
hereby grants Licensee a non-exclusive, non-transferable, royalty-free,
world-wide license to reproduce, analyze, test, perform and/or display
publicly, prepare derivative works, distribute, and otherwise use the
Software alone or in any derivative version, provided, however, that
CNRI's License Agreement and CNRI's notice of copyright, i.e.,
"Copyright 1996-1999 Corporation for National Research Initiatives;
All Rights Reserved" are both retained in the Software, alone or in any
derivative version prepared by Licensee.

Alternatively, in lieu of CNRI's License Agreement, Licensee may
substitute the following text (omitting the quotes), provided, however,
that such text is displayed prominently in the Software alone or in any
derivative version prepared by Licensee: "JPython (Version 1.1.x) is
made available subject to the terms and conditions in CNRI's License
Agreement. This Agreement may be located on the Internet using the
following unique, persistent identifier (known as a handle):
1895.22/1006. The License may also be obtained from a proxy server on
the Web using the following URL: http://hdl.handle.net/1895.22/1006."

3. In the event Licensee prepares a derivative work that is based on or
incorporates the Software or any part thereof, and wants to make the
derivative work available to the public as provided herein, then
Licensee hereby agrees to indicate in any such work, in a prominently
visible way, the nature of the modifications made to CNRI's Software.
4. Licensee may not use CNRI trademarks or trade name, including JPython or CNRI, in a trademark sense to endorse or promote products or services of Licensee, or any third party. Licensee may use the mark JPython in connection with Licensee's derivative versions that are based on or incorporate the Software, but only in the form "JPython-based ________________," or equivalent.

5. CNRI is making the Software available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

6. CNRI SHALL NOT BE LIABLE TO LICENSEE OR OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY SO THE ABOVE DISCLAIMER MAY NOT APPLY TO LICENSEE.

7. This License Agreement may be terminated by CNRI (i) immediately upon written notice from CNRI of any material breach by the Licensee, if the nature of the breach is such that it cannot be promptly remedied; or (ii) sixty (60) days following notice from CNRI to Licensee of a material remediable breach, if Licensee has not remedied such breach within that sixty-day period.

8. This License Agreement shall be governed by and interpreted in all respects by the law of the State of Virginia, excluding conflict of law provisions. Nothing in this Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee.

9. By clicking on the "ACCEPT" button where indicated, or by installing, copying or otherwise using the Software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

[ACCEPT BUTTON]

B. HISTORY OF THE SOFTWARE
=====================================

JPython was created in late 1997 by Jim Hugunin. Jim was also the primary developer while he was at CNRI. In February 1999 Barry Warsaw took over as primary developer and released JPython version 1.1.
In October 2000 Barry helped move the software to SourceForge where it was renamed to Jython. Jython 2.0 and 2.1 were developed under the Jython specific license below.

From the 2.2 release on, Jython contributors have signed Python Software Foundation contributor agreements and releases are covered under the Python Software Foundation license version 2.

The standard library is covered by the Python Software Foundation license as well. See the Lib/LICENSE file for details.

The zxJDBC package was written by Brian Zimmer and originally licensed under the GNU Public License. The package is now covered by the Jython Software License.

The command line interpreter is covered by the Apache Software License. See the org/apache/LICENSE file for details.

/*--

Copyright (C) 2000-2003 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <license AT jdom DOT org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <pm AT jdom DOT org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:
  "This product includes software developed by the JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos
THIS SOFTWARE IS PROVIDED \"AS IS\" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the JDOM Project and was originally
created by Jason Hunter <jhunter AT jdom DOT org> and
Brett McLaughlin <brett AT jdom DOT org>. For more information on
the JDOM Project, please see <http://www.jdom.org/>.

*/

===========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
===========================================================================

Apache Xerces Java
Copyright 1999-2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

3Y
Z[;]
(^
(\`a
b
cdef
gh
ij)
k\-m.n/o0p1q2r3s4t5u6v7wxSETUID
1.214 htmlparser2 3.9.2 3.9.2

1.214.1 Available under license:

Copyright 2010, 2011, Chris Winberry <chris@winberry.net>. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.215 netty-codec 4.0.48.Final

1.215.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-jar/io/netty/handler/codec/MessageToMessageCodec.java
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-jar/io/netty/handler/codec/base64/Base64Encoder.java
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-jar/io/netty/handler/codec/string/StringDecoder.java
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-jar/io/netty/handler/codec/ByteToMessageDecoder.java
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources
No license file was found, but licenses were detected in source scan.
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Written by Robert Harder and released to the public domain, as explained at http://creativecommons.org/licenses/publicdomain

Enumeration of supported Base64 dialects.

The internal lookup tables in this class has been derived from Robert Harder's Public Domain Base64 Encoder/Decoder.

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

A decoder that splits the received ByteBufs dynamically by the value of the length field in the message. It is particularly useful when you decode a binary message which has an integer header field that represents the length of the message body or the whole message.

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
that it can decode any message with a length field, which is often seen in proprietary client-server protocols. Here are some example that will give you the basic idea on which option does what.

* ch3>2 bytes length field at offset 0, do not strip header</h3>

* The value of the length field in this example is <tt>12 (0x0C)</tt> which represents the length of "HELLO, WORLD". By default, the decoder assumes that the length field represents the number of the bytes that follows the length field. Therefore, it can be decoded with the simplistic parameter combination.

* <pre>
   * lengthFieldOffset = 0
   * lengthFieldLength = 2
   * lengthAdjustment = 0
   * initialBytesToStrip = 0 (= do not strip header)
   *
   * BEFORE DECODE (14 bytes)         AFTER DECODE (14 bytes)
   +--------+----------------+      +--------+----------------+
   | Length | Actual Content |----->| Length | Actual Content |
   | 0x000C | "HELLO, WORLD" |      | 0x000C | "HELLO, WORLD" |
   +--------+----------------+      +--------+----------------+
   *
   * </pre>

* ch3>2 bytes length field at offset 0, strip header</h3>

* Because we can get the length of the content by calling <tt>@link ByteBuf#readableBytes()</tt>, you might want to strip the length field by specifying <tt>initialBytesToStrip</tt>. In this example, we specified <tt>2</tt>, that is same with the length of the length field, to strip the first two bytes.

* <pre>
   * lengthFieldOffset = 0
   * lengthFieldLength = 2
   * lengthAdjustment = 0
   * initialBytesToStrip = 2 (= the length of the Length field)
   *
   * BEFORE DECODE (14 bytes)         AFTER DECODE (12 bytes)
   +--------+----------------+      +----------------+
   | Length | Actual Content |----->| Actual Content |
   | 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
   +--------+----------------+      +----------------+
   *
   * </pre>

* ch3>2 bytes length field at offset 0, do not strip header, the length field represents the length of the whole message</h3>

* In most cases, the length field represents the length of the message body
only, as shown in the previous examples. However, in some protocols, the length field represents the length of the whole message, including the message header. In such a case, we specify a non-zero <tt>lengthAdjustment</tt>. Because the length value in this example message is always greater than the body length by <tt>2</tt>, we specify <tt>-2</tt> for compensation.

* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 2
* <b>lengthAdjustment</b>    = <b>-2</b> (= the length of the Length field)
* initialBytesToStrip = 0
*
* BEFORE DECODE (14 bytes)         AFTER DECODE (14 bytes)
* +--------+----------------+      +--------+----------------+
* | Length | Actual Content |----->| Length | Actual Content |
* | 0x000E | "HELLO, WORLD" |      | 0x000E | "HELLO, WORLD" |
* +--------+----------------+      +--------+----------------+
* </pre>

* <h3>3 bytes length field at the end of 5 bytes header, do not strip header</h3>

* The following message is a simple variation of the first example. An extra header value is prepended to the message. <tt>lengthAdjustment</tt> is zero again because the decoder always takes the length of the prepended data into account during frame length calculation.

* <pre>
* <b>lengthFieldOffset</b>   = <b>2</b> (= the length of Header 1)
* <b>lengthFieldLength</b>   = <b>3</b>
* lengthAdjustment    = 0
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* | Header 1 |  Length  | Actual Content |----->| Header 1 |  Length  | Actual Content |
* |  0xCAFE  | 0x00000C | "HELLO, WORLD" |      |  0xCAFE  | 0x00000C | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>

* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>

* This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive <tt>lengthAdjustment</tt> so that the decoder counts the extra header into the frame length calculation.

* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                        AFTER DECODE (17 bytes)
* +-------------------------------+-------------------+
* | Length | Header 1 | Actual Content |----->| Length | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE | "HELLO, WORLD" |      | 0x00000C | 0xCAFE | "HELLO, WORLD" |
* +-------------------------------+-------------------+
* </pre>
*
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header,
* strip the first header field and the length field</h3>
*
* This is a combination of all the examples above. There are the prepended
* header before the length field and the extra header after the length field.
* The prepended header affects the <tt>lengthFieldOffset</tt> and the extra
* header affects the <tt>lengthAdjustment</tt>. We also specified a non-zero
* <tt>initialBytesToStrip</tt> to strip the length field and the prepended
* header from the frame. If you don't want to strip the prepended header, you
* could specify <tt>0</tt> for <tt>initialBytesToSkip</tt>.
* <pre>
* lengthFieldOffset   = 1 (= the length of HDR1)
* lengthFieldLength   = 2
* &lt;b&gt;lengthAdjustment&lt;/b&gt;    = &lt;b&gt;1&lt;/b&gt; (= the length of HDR2)
* &lt;b&gt;initialBytesToStrip&lt;/b&gt; = &lt;b&gt;3&lt;/b&gt; (= the length of HDR1 + LEN)
*
* BEFORE DECODE (16 bytes)                        AFTER DECODE (13 bytes)
* +------|--------|------+----------------+      +------|----------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +------|--------|------+----------------+      +------|----------------+
* </pre>
*
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header,
* strip the first header field and the length field, the length field
* represents the length of the whole message</h3>
*
* Let's give another twist to the previous example. The only difference from
* the previous example is that the length field represents the length of the
* whole message instead of the message body, just like the third example.
* We have to count the length of HDR1 and Length into <tt>lengthAdjustment</tt>.<tt>
* Please note that we don't need to take the length of HDR2 into account
* because the length field already includes the whole header length.
* <pre>
* lengthFieldOffset   = 1
* lengthFieldLength   = 2
* &lt;b&gt;lengthAdjustment&lt;/b&gt;    = &lt;b&gt;3&lt;/b&gt; (= the length of HDR1 + LEN, negative)
* &lt;b&gt;initialBytesToStrip&lt;/b&gt; = &lt;b&gt;3&lt;/b&gt;
*
* BEFORE DECODE (16 bytes)  AFTER DECODE (13 bytes)
* +--------+---------------+  +--------+---------------+
* | HDR1   | Length         | HDR2   | Actual Content |
* | 0xCA   | 0x0010         | 0xFE   | "HELLO, WORLD" |
* +--------+---------------+  +--------+---------------+
* @see LengthFieldPrepender
*/

Found in path(s):
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-
jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-
jar/io/netty/handler/codec/CodecOutputList.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2012 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Written by Robert Harder and released to the public domain, as explained at
* http://creativecommons.org/licenses/publicdomain
*/
/**
* Utility class for {@link ByteBuf} that encodes and decodes to and from
* <a href="http://en.wikipedia.org/wiki/Base64">Base64</a> notation.
* 
* The encoding and decoding algorithm in this class has been derived from
* <a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain
* Base64 Encoder/Decoder</a>.
*/

Found in path(s):
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-
jar/io/netty/handler/codec/base64/Base64.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-
jar/io/netty/handler/codec/compression/Crc32c.java
* /opt/cola/permits/1202446003_1631568114.9/0/netty-codec-4-0-48-final-sources-
jar/io/netty/handler/codec/compression/JdkZlibDecoder.java

1.216 jline 2.14.3
1.217 javassist 3.22.0-GA

1.217.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
 *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 */

Found in path(s):
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/analysis/FramePrinter.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/CodeConverter.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/convert/TransformReadField.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/rmi/RemoteRef.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/convert/TransformAfter.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/ExceptionTable.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/LongVector.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/analysis/Analyzer.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/JavassistException.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/ASTree.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/web/Webserver.java
Open Source Used In DNAC Platform Arica 980

* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/ASTList.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/reflect/CannotCreateException.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/reflect/Metaobject.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/scopedpool/ScopedClassPoolFactory.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/FieldDecl.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist.Resources.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/TypeChecker.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/scopedpool/ScopedClassPoolRepository.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/CtConstructor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/CtNewMethod.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/Parser.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/CtMethod.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/stackmap/TypeData.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/ExceptionData.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/Descriptor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/Translator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/rmi/Sample.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ProceedHandler.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/util/Proxy.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/convert/TransformNew.java
Open Source Used In DNAC Platform Arica 981

* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/convert/Transformer.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/expr/Cast.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/SignatureAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/stackmap/TypedBlock.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/convert/TransformWriteField.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/util/proxy/ProxyFactory.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/JvstCodeGen.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/RuntimeClassPath.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/CtBehavior.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/ast/Member.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/util/proxy/MethodFilter.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/analysis/MultiType.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/analysis/MultiArrayType.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/ast/Keyword.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/LocalVariableAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/TokenId.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/Loader.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/DeprecatedAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/ast/ExprTag.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/ast/BinExpr.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/tools/rmi/Proxy.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/runtime/DotClass.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/CtNewWrappedConstructor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/compiler/SyntaxError.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/LocalVariableTypeAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/convert/TransformNewClass.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/LocalVariableTypeAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/convert/TransformNewClass.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/LoaderClassPath.java
jar/javassist/bytecode/FieldInfo.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/ClassFile.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/scopedpool/SoftValueHashMap.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/JvstTypeChecker.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/web/BadHttpRequest.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/util/HotSwapAgent.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/EnclosingMethodAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/Callback.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/ClassPath.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/Dump.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/StackMapTable.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/InstanceOfExpr.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/CodeGen.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/SymbolTable.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/expr/NewArray.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/Lex.java

Open Source Used In DNAC Platform Arica 983
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/analysis/Executor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/ParameterAnnotationsAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/runtime/Cflow.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/AnnotationDefaultAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/BadBytecode.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/CallExpr.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/ByteArrayClassPath.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/stackmap/Tracer.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/Visitor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/CtNewClass.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/ClassFileWriter.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/tools/framedump.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/ClassPoolTail.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/LineNumberAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/DoubleConst.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/proxy/SecurityActions.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/convert/TransformFieldAccess.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/AccessFlag.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/InnerClassesAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/runtime/Inner.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/analysis/Type.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/compiler/ast/Pair.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/SerialVersionUID.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/AnnotationsAttribute.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4-jar/javassist/bytecode/CodeIterator.java
No license file was found, but licenses were detected in source scan.

/*
 * Javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 2004 Bill Burke. All Rights Reserved.
 *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 */
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*/

Found in path(s):
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/AnnotationMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/FloatMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/ClassMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/CharMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/ShortMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/EnumMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/IntegerMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/ArrayMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/ByteMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/StringMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/MemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/MemberValueVisitor.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/LongMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/DoubleMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/BooleanMemberValue.java
* /opt/cola/permits/1136405427_1613762483.76/0/javassist-3-22-0-ga-sources-4.jar/javassist/bytecode/annotation/Annotation.java

1.218 google-http-client-xml 1.20.0
1.218.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.219 ini 2.0.0

1.219.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.220 @babel/plugin-proposal-async-generator-functions 7.16.4

1.220.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.221 jju 1.2.1 1.2.1
1.221.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.222 binary-extensions 2.1.0
1.222.1 Available under license:
MIT License

Copyight (c) 2019 Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com), Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.223 junit 4.13.2
1.223.1 Available under license:
JUnit

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient
copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
1.224 batik-svgbrowser 1.13

1.224.1 Available under license:

The following files are from the Tango Desktop Project
(https://tango.freedesktop.org):

document-open.png
document-open.png
document-print.png
edit-find.png
go-next.png
go-previous.png
icon-general.png (originally preferences-desktop.png)
icon-language.png (originally preferences-desktop-locale.png)
icon-network.png (originally applications-internet.png)
icon-security.png (originally dialog-warning.png)
icon-stylesheet.png (originally applications-graphics.png)
media-playback-pause.png
media-playback-start.png
process-stop.png
redo.png (modified version of edit-redo.png)
system-search.png
text-html.png
undo.png (modified version of edit-undo.png)
utilities-system-monitor.png
view-refresh.png
window-new.png

The following files are from the Pasodoble Icon Theme

<table>
<thead>
<tr>
<th>File</th>
<th>Original filename</th>
</tr>
</thead>
<tbody>
<tr>
<td>zoom-in.png</td>
<td>viewmag+.png</td>
</tr>
<tr>
<td>zoom-out.png</td>
<td>viewmag-.png</td>
</tr>
</tbody>
</table>

All of these image files are licensed under the Creative Commons Attribution Share-Alike 2.5 License (http://creativecommons.org/licenses/by-sa/2.5/), which is in the LICENSE.icons.txt file.

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED
HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
b. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
c. "Licensor" means the individual or entity that offers the Work under the terms of this License.
d. "Original Author" means the individual or entity who created the Work.
e. "Work" means the copyrightable work of authorship offered under the terms of this License.
f. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
g. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.

2. Fair Use Rights.

Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant.

Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
b. to create and reproduce Derivative Works;
c. to distribute copies or phonorecords of, display publicly, perform
publicly, and perform publicly by means of a digital audio transmission
the Work including as incorporated in Collective Works;
d. to distribute copies or phonorecords of, display publicly, perform
publicly, and perform publicly by means of a digital audio transmission
Derivative Works.
e. For the avoidance of doubt, where the work is a musical composition:
   i. Performance Royalties Under Blanket Licenses. Licensor waives the
      exclusive right to collect, whether individually or via a
      performance rights society (e.g. ASCAP, BMI, SESAC), royalties for
      the public performance or public digital performance (e.g. webcast)
      of the Work.
   ii. Mechanical Rights and Statutory Royalties. Licensor waives the
       exclusive right to collect, whether individually or via a music
       rights society or designated agent (e.g. Harry Fox Agency),
       royalties for any phonorecord You create from the Work ("cover
       version") and distribute, subject to the compulsory license created
       by 17 USC Section 115 of the US Copyright Act (or the equivalent in
       other jurisdictions).
f. Webcasting Rights and Statutory Royalties. For the avoidance of doubt,
where the Work is a sound recording, Licensor waives the exclusive right
to collect, whether individually or via a performance-rights society (e.g.
SoundExchange), royalties for the public digital performance (e.g.
webcast) of the Work, subject to the compulsory license created by 17 USC
Section 114 of the US Copyright Act (or the equivalent in other
jurisdictions).

The above rights may be exercised in all media and formats whether now known or
hereafter devised. The above rights include the right to make such modifications
as are technically necessary to exercise the rights in other media and formats.
All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions.

The license granted in Section 3 above is expressly made subject to and limited
by the following restrictions:

a. You may distribute, publicly display, publicly perform, or publicly
digitally perform the Work only under the terms of this License, and You
must include a copy of, or the Uniform Resource Identifier for, this
License with every copy or phonorecord of the Work You distribute,
publicly display, publicly perform, or publicly digitally perform. You may
not offer or impose any terms on the Work that alter or restrict the terms
of this License or the recipients’ exercise of the rights granted
hereunder. You may not sublicense the Work. You must keep intact all
notices that refer to this License and to the disclaimer of warranties.
You may not distribute, publicly display, publicly perform, or publicly
digitally perform the Work with any technological measures that control
access or use of the Work in a manner inconsistent with the terms of this
license Agreement. The above applies to the Work as incorporated in a
Collective Work, but this does not require the Collective Work apart
from the Work itself to be made subject to the terms of this License. If You
create a Collective Work, upon notice from any Licensor You must, to the
extent practicable, remove from the Collective Work any credit as required
by clause 4(c), as requested. If You create a Derivative Work, upon notice
from any Licensor You must, to the extent practicable, remove from the
Derivative Work any credit as required by clause 4(c), as requested.
b. You may distribute, publicly display, publicly perform, or publicly
digitally perform a Derivative Work only under the terms of this License,
a later version of this License with the same License Elements as this
License, or a Creative Commons iCommons license that contains the same
License Elements as this License (e.g. Attribution-ShareAlike 2.5 Japan).
You must include a copy of, or the Uniform Resource Identifier for, this
License or other license specified in the previous sentence with every
copy or phonorecord of each Derivative Work You distribute, publicly
display, publicly perform, or publicly digitally perform. You may not
offer or impose any terms on the Derivative Works that alter or restrict
the terms of this License or the recipients' exercise of the rights
granted hereunder, and You must keep intact all notices that refer to
this License and to the disclaimer of warranties. You may not distribute,
publicly display, publicly perform, or publicly digitally perform the
Derivative Work with any technological measures that control access or
use of the Work in a manner inconsistent with the terms of this License
Agreement. The above applies to the Derivative Work as incorporated in
a Collective Work, but this does not require the Collective Work apart
from the Derivative Work itself to be made subject to the terms of this
License.
c. If you distribute, publicly display, publicly perform, or publicly
digitally perform the Work or any Derivative Works or Collective Works,
You must keep intact all copyright notices for the Work and provide,
reasonable to the medium or means You are utilizing: (i) the name of the
Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if
the Original Author and/or Licensor designate another party or parties
(e.g. a sponsor institute, publishing entity, journal) for attribution
in Licensor's copyright notice, terms of service or by other reasonable
means, the name of such party or parties; the title of the Work if
supplied; to the extent reasonably practicable, the Uniform Resource
Identifier, if any, that Licensor specifies to be associated with the
Work, unless such URI does not refer to the copyright notice or licensing
information for the Work; and in the case of a Derivative Work, a credit
identifying the use of the Work in the Derivative Work (e.g., "French
translation of the Work by Original Author," or "Screenplay based on
original Work by Original Author"). Such credit may be implemented in any
reasonable manner; provided, however, that in the case of a Derivative
Work or Collective Work, at a minimum such credit will appear where any
other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE MATERIALS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.


EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this
License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for Standardization for the definition of character entities used in the software’s documentation.

This product includes images from the Tango Desktop Project (http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme (http://www.jesusda.com/projects/pasodoble).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed.
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.225 vm2 3.8.3 3.8.3
1.225.1 Available under license:
MIT License

Copyright (c) 2014-2019 Patrik Simek and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.226 ip-address 5.8.9 5.8.9

1.226.1 Available under license:
Copyright (C) 2011 by Beau Gunderson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from
the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.228 string-width 4.2.2 4.2.2

1.228.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.229 os-locale 1.4.0 1.4.0

1.229.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.230 cliui 6.0.0

1.230.1 Available under license:

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.231 dot-prop 5.3.0

1.231.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.232 jetty-servlet 9.3.27.v20190418

1.233 xml2js 0.4.23 0.4.23

1.233.1 Available under license:

Copyright 2010, 2011, 2012, 2013. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.234 protostuff-api 1.7.1

1.234.1 Available under license:

No license file was found, but licenses were detected in source scan.

//Licensed under the Apache License, Version 2.0 (the "License");
//you may not use this file except in compliance with the License.
//You may obtain a copy of the License at
//http://www.apache.org/licenses/LICENSE-2.0
//distributed under the License is distributed on an "AS IS" BASIS,
//Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/EnumLite.java
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/UninitializedMessageException.java
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/ByteString.java
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/WireFormat.java

No license file was found, but licenses were detected in source scan.

//Licensed under the Apache License, Version 2.0 (the "License");
//you may not use this file except in compliance with the License.
//You may obtain a copy of the License at
//http://www.apache.org/licenses/LICENSE-2.0
//distributed under the License is distributed on an "AS IS" BASIS,
*/
* Copyright (C) 2011 The Guava Authors
* * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* * http://www.apache.org/licenses/LICENSE-2.0
* * Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/UnsignedNumberUtil.java
No license file was found, but licenses were detected in source scan.

//All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are met:
// 1. Redistributions of source code must retain the above copyright notice,
//    this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright notice,
//    this list of conditions and the following disclaimer in the documentation
// 3. Neither the name of protostuff nor the names of its contributors may be used

Found in path(s):
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/Tag.java
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/Exclude.java
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/Morph.java
No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/ListAdapter.java
No license file was found, but licenses were detected in source scan.

//Licensed under the Apache License, Version 2.0 (the "License");
//you may not use this file except in compliance with the License.
//You may obtain a copy of the License at
//http://www.apache.org/licenses/LICENSE-2.0
//distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1202446559_1631568036.94/0/protostuff-api-1-7-1-sources-jar/io/protostuff/B64Code.java
1.235 json-schema 0.2.3 0.2.3

1.235.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.236 require-directory 2.1.1 2.1.1

1.236.1 Available under license:

The MIT License (MIT)

Copyright (c) 2011 Troy Goode <troygoode@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.237 semver 7.3.5

1.237.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.238 jersey-client 2.32

1.238.1 Available under license:

  # Notice for Jersey
  This content is produced and maintained by the Eclipse Jersey project.

  * Project home: https://projects.eclipse.org/projects/ee4j.jersey

  ## Trademarks
  Eclipse Jersey is a trademark of the Eclipse Foundation.

  ## Copyright
  All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

  ## Declared Project Licenses
  This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

  SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

  ## Source Code
  The project maintains the following source code repositories:

  * https://github.com/eclipse-ee4j/jersey

  ## Third-party Content

  Angular JS, v1.6.6
  * License MIT (http://www.opensource.org/licenses/mit-license.php)
  * Project: http://angularjs.org
  * Copyright: (c) 2010-2017 Google, Inc.

  aopalliance Version 1
  * License: all the source code provided by AOP Alliance is Public Domain.
  * Copyright: Material in the public domain is not protected by copyright
Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
* by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.
"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby
grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice
"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

```markdown
## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These
restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis
or for a fee, you must give the recipients all the rights that you have.
You must make sure that they, too, receive or can get the source code.
And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software patents.
We wish to avoid the danger that redistributors of a free program will
individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must
be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a
notice placed by the copyright holder saying it may be distributed under
the terms of this General Public License. The "Program", below, refers
to any such program or work, and a "work based on the Program" means
either the Program or any derivative work under copyright law: that is
to say, a work containing the Program or a portion of it, either
verbatim or with modifications and/or translated into another language.
(Hereinafter, translation is included without limitation in the term
"modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of running
the Program is not restricted, and the output from the Program is
covered only if its contents constitute a work based on the Program
(independent of having been made by running the Program). Whether that
is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source
code as you receive it, in any medium, provided that you conspicuously
and appropriately publish on each copy an appropriate copyright notice
and disclaimer of warranty; keep intact all the notices that refer to
this License and to the absence of any warranty; and give any other
recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of
it, thus forming a work based on the Program, and copy and distribute
such modifications or work under the terms of Section 1 above, provided
that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
      stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
      whole or in part contains or is derived from the Program or any part
      thereof, to be licensed as a whole at no charge to all third parties
      under the terms of this License.

   c) If the modified program normally reads commands interactively
      when run, you must cause it, when started running for such
      interactive use in the most ordinary way, to print or display an
      announcement including an appropriate copyright notice and a notice
      that there is no warranty (or else, saying that you provide a
      warranty) and that users may redistribute the program under these
      conditions, and telling the user how to view a copy of this License.
      (Exception: if the Program itself is interactive but does not
      normally print such an announcement, your work based on the Program
      is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program, and
can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on
the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.239 jquery 3.6.0 3.6.0

1.239.1 Available under license :

Copyright OpenJS Foundation and other contributors, https://openjsf.org/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The BSD License)

Copyright (c) 2010-2017, Christian Johansen, christian@cjohansen.no
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Christian Johansen nor the names of his contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright JS Foundation and other contributors, https://js.foundation

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/qunitjs/qunit
The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules directory are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above. Copyright JS Foundation and other contributors, https://js.foundation/

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/sizzle

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.240 slf4j-log4j 2.14.0
1.240.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.241 1to2 1.0.0 1.0.0

1.241.1 Available under license:

The MIT License (MIT)
Copyright (c) 2018 NAN contributors

NAN contributors listed at https://github.com/nodejs/nan#contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.242 babel-code-frame 6.26.0 6.26.0

1.242.1 Available under license:

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.243 strip-bom 3.0.0

1.243.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.244 safer-buffer 2.1.2 2.1.2
1.244.1 Available under license:
MIT License

Copyright (c) 2018 Nikita Skovoroda <chalkerx@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"). to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.245 objenesis 2.5.1

1.245.1 Available under license:

// ------------------------------------------------------------------
// NOTICE file corresponding to the section 4d of The Apache License,
// Version 2.0, in this case for Objenesis
// ------------------------------------------------------------------

Objenesis
Copyright 2006-2017 Joe Walnes, Henri Tremblay, Leonardo Mesquita

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.246 jetty-io 9.4.41.v20210516
1.246.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

===============================================

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

===============================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and
* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
  * org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
  * org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
  * org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
  https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

  * javax.servlet:javax.servlet-api
  * javax.annotation:javax.annotation-api
  * javax.transaction:javax.transaction-api
  * javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

  * java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
  http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2
The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.247 normalize-path 3.0.0 3.0.0
1.247.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.248 flagged-respawn 1.0.1 1.0.1
1.248.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2018 Tyler Kellen <tyler@sleekcode.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.249 slash 2.0.0
1.249.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.250 jnr-posix 3.0.27
1.250.1 Available under license :
Found license 'GNU Lesser General Public License' in 'License Version 1.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of Software distributed under the License is distributed on an "AS Alternatively, the contents of this file may be used under the terms of either of the GNU General Public License Version 2 or later (the "GPL"), or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), use your version of this file under the terms of the CPL, indicate your'
1.251 validator 10.11.0 10.11.0
1.251.1 Available under license:

Copyright (c) 2018 Chris O'Hara <cohara87@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.252 ansi-styles 4.3.0 4.3.0
1.252.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.253 z-schema 3.25.1

1.253.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Martin Zagora and other contributors
https://github.com/zaggino/z-schema/graphs/contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.254 babel-runtime 6.26.0

1.254.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The "Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.255 has-symbols 1.0.1

1.255.1 Available under license:

MIT License

Copyright (c) 2016 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.256 is-negative-zero 2.0.0
1.256.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.257 xmlbuilder 8.2.2 8.2.2
1.257.1 Available under license :
The MIT License (MIT)

Copyright (c) 2013 Ozgur Ozcitak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.258 ua-parser-js 0.7.18 0.7.18

1.258.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.259 parseuri 0.0.5 0.0.5

1.259.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Gal Koren

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.260 rxjs 6.6.3 6.6.3
1.260.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work.
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2015-2018 Google, Inc., Netflix, Inc., Microsoft Corp. and contributors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.261 babel-plugin-proposal-nullish-coalescing-operator 7.16.0

1.261.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.262 domutils 1.7.0 1.7.0
1.262.1 Available under license :
Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.263 @babel/helper-validator-identifier 7.10.4
1.263.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE 
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION 
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.264 drill-format-spss 1.19.0
1.264.1 Available under license :

Drill : Contrib : Format : SPSS
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, 
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by 
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all 
other entities that control, are controlled by, or are under common 
control with that entity. For the purposes of this definition, 
"control" means (i) the power, direct or indirect, to cause the 
direction or management of such entity, whether by contract or 
otherwise, or (ii) ownership of fifty percent (50%) or more of the 
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.265 @babel/helper-define-polyfill-provider
0.3.0
1.265.1 Available under license :
MIT License

Copyright (c) 2014-present Nicol Ribaudo and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.266 es-abstract 1.18.0

1.266.1 Available under license:

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.267 commons-beanutils 1.9.4
1.267.1 Available under license:
Apache Commons BeanUtils
Copyright 2000-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.268 socket.io 2.3.0 2.3.0

1.268.1 Available under license :
(The MIT License)

Copyright (c) 2014-2018 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.269 escope 3.6.0
1.269.1 Available under license:
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.270 ws 7.3.1 7.3.1

1.270.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011 Einar Otto Stangvik <einaros@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.271 json-stable-stringify 1.0.1

1.271.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.272 wrap-ansi 5.1.0

1.272.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.273 intl-format-cache 2.1.0 2.1.0

1.273.1 Available under license:

Copyright 2014 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.274 user-home 1.1.1

1.274.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.275 libflac 1.3.1

1.275.1 Available under license:

GNU Free Documentation License
Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.
1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file
format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF, and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept
compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under
the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retile any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original
Open Source Used In DNAC Platform Arica

author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may
distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents
To use this License in a document you have written, include a copy of
the License in the document and put the following copyright and
license notices just after the title page:

Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts,
replace the "with...Texts." line with this:

   with the Invariant Sections being LIST THEIR TITLES, with the
   Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other
combination of the three, merge those two alternatives to suit the
situation.

If your document contains nontrivial examples of program code, we
recommend releasing these examples in parallel under your choice of
free software license, such as the GNU General Public License,
to permit their use in free software.

GNU LESSER GENERAL PUBLIC LICENSE
   Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the

Open Source Used In DNAC Platform Arica  1101
Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
FLAC is a free codec in the fullest sense. This page explicitly states all that you may do with the format and software.

The FLAC and Ogg FLAC formats themselves, and their specifications, are fully open to the public to be used for any purpose (the FLAC project reserves the right to set the FLAC specification and certify compliance). They are free for commercial or noncommercial use. That means that commercial developers may independently write FLAC or Ogg FLAC software which is compatible with the specifications for no charge and without restrictions of any kind. There are no licensing fees or royalties of any kind for use of the formats or their specifications, or for distributing, selling, or streaming media in the FLAC or Ogg FLAC formats.

The FLAC project also makes available software that implements the formats, which is distributed according to Open Source licenses as follows:

The reference implementation libraries are licensed under the New BSD License. In simple terms, these libraries may be used by any application, Open or proprietary, linked or incorporated in whole, so long as acknowledgement is made to Xiph.org Foundation when using the source code in whole or in derived works. The Xiph License is free enough that the libraries have been used in commercial products to implement FLAC, including in the firmware of hardware devices where other Open Source licenses can be problematic. In the source code these libraries are called libFLAC and libFLAC++.

The rest of the software that the FLAC project provides is licensed under the GNU General Public License (GPL). This software includes various utilities for converting files to and from FLAC format, plugins for audio players, et cetera. In general, the GPL allows redistribution as long as derived works are also made available in source code form according to compatible terms.

Neither the FLAC nor Ogg FLAC formats nor any of the implemented encoding/decoding methods are covered by any known patent.

FLAC is one of a family of codecs of the Xiph.org Foundation, all created according to the same free ideals. For some other codecs' descriptions of the Xiph License see the Speex and Vorbis license pages.

If you would like to redistribute parts or all of FLAC under different terms, contact the FLAC-dev mailinglist.

Contact the FLAC-dev mailinglist.
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable
      source code, which must be distributed under the terms of Sections
      1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three
      years, to give any third party, for a charge no more than your
      cost of physically performing source distribution, a complete
      machine-readable copy of the corresponding source code, to be
      distributed under the terms of Sections 1 and 2 above on a medium
      customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer
      to distribute corresponding source code. (This alternative is
      allowed only for noncommercial distribution and only if you
      received the program in object code or executable form with such
      an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Copyright (C) 2000-2009 Josh Coalson
Copyright (C) 2011-2016 Xiph.Org Foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.276 jetty-util-ajax 9.4.34.v20201102

1.276.1 Available under license:

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates’
from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program” means the Contributions distributed in accordance with this
Agreement.

"Recipient” means anyone who receives the Program under this Agreement,
including all Contributors.
2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants
      Recipient a non-exclusive, worldwide, royalty-free copyright license to
      reproduce, prepare derivative works of, publicly display, publicly
      perform, distribute and sublicense the Contribution of such Contributor,
      if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants
      Recipient a non-exclusive, worldwide, royalty-free patent license under
      Licensed Patents to make, use, sell, offer to sell, import and otherwise
      transfer the Contribution of such Contributor, if any, in source code and
      object code form. This patent license shall apply to the combination of
      the Contribution and the Program if, at the time the Contribution is
      added by the Contributor, such addition of the Contribution causes such
      combination to be covered by the Licensed Patents. The patent license
      shall not apply to any other combinations which include the Contribution.
      No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses
      to its Contributions set forth herein, no assurances are provided by any
      Contributor that the Program does not infringe the patent or other
      intellectual property rights of any other entity. Each Contributor
      disclaims any liability to Recipient for claims brought by any other
      entity based on infringement of intellectual property rights or
      otherwise. As a condition to exercising the rights and licenses granted
      hereunder, each Recipient hereby assumes sole responsibility to secure
      any other intellectual property rights needed, if any. For example, if a
      third party patent license is required to allow Recipient to distribute
      the Program, it is Recipient's responsibility to acquire that license
      before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient
      copyright rights in its Contribution, if any, to grant the copyright
      license set forth in this Agreement.

3. REQUIREMENTS

   A Contributor may choose to distribute the Program in object code form under
   its own license agreement, provided that:

   a) it complies with the terms and conditions of this Agreement; and
   b) its license agreement:
      i) effectively disclaims on behalf of all Contributors all warranties
         and conditions, express and implied, including warranties or
         conditions of title and non-infringement, and implied warranties or
         conditions of merchantability and fitness for a particular purpose;
      ii) effectively excludes on behalf of all Contributors all liability for
          damages, including direct, indirect, special, incidental and
          consequential damages, such as lost profits;
      iii) states that any provisions which differ from this Agreement are
          offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
  a) promptly notify the Commercial Contributor in writing of such claim, and
  b) allow the Commercial Contributor to control, and cooperate with the
     Commercial Contributor in, the defense and any related settlement
     negotiations. The Indemnified Contributor may participate in any such claim at
     its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a
court requires any other Contributor to pay any damages as a result, the
Commercial Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html
and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-------
The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.277 eslint-module-utils 2.7.1
1.277.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Ben Mosher

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.278 @babel/plugin-transform-modules-commonjs 7.16.0
1.278.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.279 aggregate-error 3.1.0

1.279.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.280 supports-color 7.2.0

1.280.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.281 fresh 0.5.2 0.5.2

1.281.1 Available under license:
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2016-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.282 intl-relativeformat 2.1.0 2.1.0
1.282.1 Available under license:

Copyright 2014 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------------------------

Pluralization rules built from
https://github.com/papandreou/node-cldr
which is licensed under the BSD license and has the following license:

Copyright (c) 2012, Andreas Lind Petersen
All rights reserved.

See the following for more details:
https://github.com/papandreou/node-cldr/blob/master/LICENSE

1.283 groovy-testng 3.0.8

1.283.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  * /opt/cola/permits/1261652993_1643306451.19/0/groovy-testng-3-0-8-sources-
    jar/org/apache/groovy/plugin/testng/TestNgRunner.java
  * /opt/cola/permits/1261652993_1643306451.19/0/groovy-testng-3-0-8-sources-
    jar/org/codehaus/groovy/testng/TestNgRunner.java

No license file was found, but licenses were detected in source scan.

#  Licensed to the Apache Software Foundation (ASF) under one
# or more contributor license agreements. See the NOTICE file
# distributed with this work for additional information
# regarding copyright ownership. The ASF licenses this file
# to you under the Apache License, Version 2.0 (the
# "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at
#
#    http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing,
# software distributed under the License is distributed on an
# "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
# KIND, either express or implied. See the License for the
# specific language governing permissions and limitations
# under the License.
#

org.apache.groovy.plugin.testng.TestNgRunner

Found in path(s):
  * /opt/cola/permits/1261652993_1643306451.19/0/groovy-testng-3-0-8-sources-jar/META-INF/services/org.apache.groovy.plugin.GroovyRunner
1.284 lodash 4.17.10 4.17.10

1.284.1 Available under license:

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.285 kerb-crypto 1.0.0
1.285.1 Available under license:

Kerby-kerb Crypto
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTCICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.286 parseurl 1.3.3 1.3.3

1.286.1 Available under license :

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.287 regenerator-runtime 0.13.9
1.287.1 Available under license :
MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.288 jquery 3.5.1 3.5.1
1.288.1 Available under license :
Copyright JS Foundation and other contributors, https://js.foundation/

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.
1.289 bitsyntax 0.0.4 0.0.4

1.289.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.290 underscore 1.13.1 1.13.1

1.290.1 Available under license:

Copyright (c) 2009-2021 Jeremy Ashkenas, Julian Gonggrijp, and DocumentCloud and Investigative Reporters & Editors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.291 babel-register 6.26.0 6.26.0
1.291.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.292 strip-bom-buf 2.0.0 2.0.0
1.292.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.293 neo-async 2.6.2 2.6.2

1.293.1 Available under license :
MIT License

Copyright (c) 2014-2018 Suguru Motegi
Based on Async.js, Copyright Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.294 isemail 1.2.0 1.2.0

1.294.1 Available under license :
Copyright 2008-2011, Dominic Sayers
Copyright 2013-2014, GlobeSherpa
Copyright 2014-2015, Eli Skeggs

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Dominic Sayers nor the names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.295 batik-svvggen 1.13
1.295.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svvggen-1-13-sources-1-
jar/org/apache/batik/svggen/CachedImageHandlerJPEGEncoder.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svvggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/GlyfCompositeComp.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svvggen-1-13-sources-1-
Open Source Used In DNAC Platform Arica  1150
Open Source Used In DNAC Platform Arica

* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/ErrorConstants.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttributeMap.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/ImageHandlerJPEGEncoder.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/SVGFont.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGPolygon.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/ScriptList.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/ClassDefFormat1.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGShape.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/KernSubtable.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGPaint.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SwingSVGPrettyPrint.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/AbstractSVGFilterConverter.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/ErrorHandler.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttribute.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttributeMap.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGColor.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/Device.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/Point.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/LigatureSubst.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/LigatureSubst.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGColor.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/Glyph.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/table/Program.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttribute.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttributeMap.java
* /opt/cola/permits/1146690661_1615697247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/SVGAttribute.java
jar/org/apache/batik/svggen/SVGComposite.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/HmtxTable.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/DefaultStyleHandler.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/DOMTreeManager.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/GlyphDescription.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGDescriptor.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/Lookup.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGGraphicContext.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/AbstractImageHandlerEncoder.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/DefaultExtensionHandler.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/GlyfCompositeDescriptor.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGFont.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/LigatureSet.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/CmapFormat4.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGHintsDescriptor.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/FpgmTable.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGBufferedImageOp.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/font/table/Os2Table.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGCompositeDescriptor.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
jar/org/apache/batik/svggen/SVGCUSTOMBUFFEREDIMAGEOP.java
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-
Open Source Used In DNAC Platform Arica  1154
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

    # The ASF licenses this file to You under the Apache License, Version 2.0
    # (the "License"); you may not use this file except in compliance with
    # the License. You may obtain a copy of the License at
    #     http://www.apache.org/licenses/LICENSE-2.0
    # distributed under the License is distributed on an "AS IS" BASIS,

    Found in path(s):
* /opt/cola/permits/1146690661_1616597247.74/0/batik-svggen-1-13-sources-1-jar/org/apache/batik/svggen/font/resources/Messages.properties

1.296 commander 1.2.0 1.2.0
1.296.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.297 lodash-union 4.6.0 4.6.0
1.297.1 Available under license :

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.298 mongodb 2.2.36 2.2.36

1.298.1 Available under license:
Mongodb-core uses third-party libraries or other resources that may be distributed under licenses different than the Mongodb-core software.

In the event that we accidentally failed to list a required notice, please bring it to our attention through any of the ways detailed here:

mongodb-dev@googlegroups.com

The attached notices are provided for information only.

For any licenses that require disclosure of source, sources are available at https://github.com/mongodb/node-mongodb-native.

1) License Notice for nan.h

The MIT License (MIT)

Copyright (c) 2016 NAN contributors
NAN contributors listed at <https://github.com/nodejs/nan#contributors>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to
do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

Open Source Used In DNAC Platform Arica 1161
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.299 antlr-runtime 4.8-1

1.299.1 Available under license:

No license file was found, but licenses were detected in source scan.

~ Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/META-INF/maven/org.antlr/antlr4-runtime/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.
 * Use of this file is governed by the BSD 3-clause license that
 * can be found in the LICENSE.txt file in the project root.
 */
Open Source Used In DNAC Platform Arica  1164

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/Array2DHashSet.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/dfa/DFASerializer.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/FailedPredicateException.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/InterpreterDataReader.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/EqualityComparator.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/LoopEndState.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/Predicate.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/ANTLRInputStream.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/pattern/ParseTreePatternMatcher.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/RangeTransition.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/LookaheadEventInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/pattern/TokenTagToken.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/CodePointBuffer.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/RuleContextWithAltNum.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/IntegerStack.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/ActionTransition.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/LexerActionExecutor.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/OrderedATNConfigSet.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/BaseErrorListener.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/LexerSkipAction.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/DecisionEventInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/RuntimeMetaData.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/InterpreterRuleContext.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/Recognizer.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathTokenElement.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/LexerNoViableAltException.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/LexerModeAction.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/CommonTokenStream.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/SetTransition.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/dfa/DFAState.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/ANTLRFileStream.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/ATNDeserializationOptions.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/pattern/ParseTreeMatch.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/DecisionInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathTokenAnywhereElement.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/TokenStreamRewriter.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/ErrorNode.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/BlockStartState.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathRuleElement.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/AmbiguityInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/ErrorNode.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/LexerInterpreter.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathTokenElement.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/atn/PredicateEvalInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/pattern/ParseTreeMatch.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/ATNType.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/LexerPushModeAction.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/pattern/RuleTagToken.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/PlusLoopbackState.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/TokenStream.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/LexerTypeAction.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/ParseTreeListener.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/ProxyErrorListener.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/PredictionContextCache.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/ParseTreeVisitor.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/UnbufferedCharStream.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/ATNConfig.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/ConsoleErrorListener.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/SyntaxTree.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathLexer.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/CharStreams.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tree/RuleStopState.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/IterativeParseTreeWalker.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/ErrorInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/tn/ContextSensitivityInfo.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/misc/IntSet.java
* /opt/ws_local/PERMITS_SQL/1092523039_1600741420.05/0/antlr4-runtime-4-8-1-sources-jar/org/antlr/v4/runtime/TokenSource.java
1.300 babel-plugin-polyfill-corejs3 0.4.0

1.300.1 Available under license:

MIT License

Copyright (c) 2014-present Nicol Ribaudo and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.301 keygrip 1.0.3 1.0.3
1.301.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011-2014 Jed Schmidt <where@jed.is> (http://jedschmidt.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.302.1 Available under license:
Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS,
1.303 rimraf 2.7.1 2.7.1

1.303.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.304 read-all-stream 1.0.2 1.0.2

1.304.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.305 harfbuzz 1.0.1-1ubuntu0.1

1.305.1 Available under license:

The contents of this directory are licensed under the following terms:

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

HarfBuzz is licensed under the so-called "Old MIT" license. Details follow.

For parts of HarfBuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stribley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

CC BY-NC-SA 3.0
http://www.crulp.org/software/license/CreativeCommons.html
https://creativecommons.org/licenses/by-nc-sa/3.0/
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0
Upstream-Name: HarfBuzz
Upstream-Contact: Behdad Esfahbod
Source: http://www.freedesktop.org/wiki/Software/HarfBuzz

Files: *
2012,2013,2015 Mozilla Foundation
2011 Codethink Limited
2008,2010 Nokia Corporation and/or its subsidiary(-ies)
2009 Keith Stribley
2009 Martin Hosken and SIL International
2007 Chris Wilson
2006 Behdad Esfahbod
2005 David Turner
1998-2004 David Turner and Werner Lemberg
License: MIT

Files: debian/*
Copyright: 2012-2015 (Ahmed El-Mahmoudy) <aelmahmoudy@users.sourceforge.net>
License: MIT

License: MIT
Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
Copyright (c) 2010 Red Hat Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.306 base64-js 1.3.1 1.3.1

1.306.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jameson Little

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.307 home-or-tmp 2.0.0 2.0.0
1.307.1 Available under license:
   The MIT License (MIT)

   Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

   Permission is hereby granted, free of charge, to any person obtaining a copy of
   this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
   all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
   THE SOFTWARE.

1.308 de.huxhorn.lilith.sender 0.9.44
1.308.1 Available under license:
   GNU LESSER GENERAL PUBLIC LICENSE
       Version 3, 29 June 2007

   Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
   Everyone is permitted to copy and distribute verbatim copies
   of this license document, but changing it is not allowed.

   This version of the GNU Lesser General Public License incorporates
   the terms and conditions of version 3 of the GNU General Public
   License, supplemented by the additional permissions listed below.

   0. Additional Definitions.

   As used herein, "this License" refers to version 3 of the GNU Lesser
   General Public License, and the "GNU GPL" refers to version 3 of the
   GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.309 graceful-fs 4.1.11 4.1.11
1.309.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter, Ben Noordhuis, and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.310 lodash-difference 4.5.0 4.5.0
1.310.1 Available under license :
Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.311 strict-uri-encode 2.0.0 2.0.0

1.311.1 Available under license:

The MIT License (MIT)

Copyright (c) Kevin Martesson <kevinmartesson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1.312 path-root 0.1.1 0.1.1

1.312.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.313 es6-symbol 3.1.1

1.313.1 Available under license :

Copyright (C) 2013-2015 Mariusz Nowak (www.medikoo.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.314 typedarray-to-buffer 3.1.5

1.314.1 Available under license :

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.315 jackson-annotations 2.12.1

1.315.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.316 is-plain-object 2.0.4 2.0.4
1.316.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.317 jackson-module-paranamer 2.10.5
1.317.1 Available under license :
This copy of Jackson JSON processor `jackson-module-paranamer` module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing
Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

### 1.318 has-value 0.3.1 0.3.1

**1.318.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.319 safe-regex 1.1.0 1.1.0

**1.319.1 Available under license:**

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.320 Iodash-map 4.6.0 4.6.0

1.320.1 Available under license:
Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.321 color-name 1.1.4 1.1.4
1.321.1 Available under license:
The MIT License (MIT)
Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.322 is-core-module 2.8.0
1.322.1 Available under license:
The MIT License (MIT)
Copyright (c) 2014 Dave Justice

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.323 jersey-container-servlet 2.5.1
1.323.1 Available under license :
CDDL+GPL License
http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.324 through 2.3.8 2.3.8
1.324.1 Available under license :
The MIT License

Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge,
to any person obtaining a copy of this software and
associated documentation files (the "Software"), to
deal in the Software without restriction, including
without limitation the rights to use, copy, modify,
merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom
the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice
shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License, Version 2.0

Copyright (c) 2011 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.325 source-map 0.6.0 0.6.0

1.325.1 Available under license:

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
1.326 hadoop-distcp 3.2.2
1.326.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of
Gson 2.2.4,
which has the following notices:

The Netty Project
=================
Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source
Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
This product optionally depends on 'Compress-LZF', a Java library for encoding and
decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression
and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression
and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression
and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data
interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate
a temporary self-signed X.509 certificate when the JVM does not provide the
equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
* license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
* http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
* license/COPYING (Apache License 2.0)
* HOMEPAGE:
* https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API which can be obtained at:

* LICENSE:
* license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
* http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
* license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
* http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
* license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
* http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
* license/LICENSE.log4j.txt (Apache License 2.0)
This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fastxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors. Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of Jetty :: Http Utility 9.3.19., 
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:
* -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and
* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2
The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/ (Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUntime Library Exception" http://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* =========================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License, ==
==  Version 2.0, in this case for the Apache Xerces Java distribution. ==
==========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.
The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.
Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--

This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
*============================================================================
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
=============================================================================
This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agree to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.  
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright 
notices and license terms. Your use of the source code for the these 
subcomponents is subject to the terms and conditions of the following 
licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract 
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL
 * nor the names of its contributors may be used to endorse or
 * promote products derived from this software without specific prior
 * written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
 * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
* ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
*/

For portions of the native implementation of slicing-by-8 CRC calculation
in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor
  the names of its contributors may be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/*++
 *
* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
*
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html
*
* Abstract: The main routine
For src/main/native/src/org/apache/hadoop/io/compress/lz4/{lz4.h,lz4.c,lz4hc.h,lz4hc.c}

/*
   LZ4 - Fast LZ compression algorithm
   Header File
   Copyright (C) 2011-2014, Yann Collet.
   BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

   Redistribution and use in source and binary forms, with or without
   modification, are permitted provided that the following conditions are
   met:

   * Redistributions of source code must retain the above copyright
     notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
     copyright notice, this list of conditions and the following disclaimer
     in the documentation and/or other materials provided with the
     distribution.

   THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
   "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
   LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
   A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
   OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
   SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
   LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
   DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
   THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
   (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
   OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

   You can contact the author at :
   - LZ4 source repository : http://code.google.com/p/lz4/
   - LZ4 public forum : https://groups.google.com/forum/#!forum/lz4c
/*

For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

re2j 1.1

--------------------------------------------------------------------------------

(RO license)

This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/fstatatopenatunlinkat.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

==============

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
--------------------------------------------------------------------------------

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
- bootstrap v3.3.6
- broccoli-asset-rev v2.4.2
- broccoli-funnel v1.0.1
- datatables v1.10.8
- emHelpers v0.5.13
- em-table v0.1.6
- ember v2.2.0
- ember-array-contains-helper v1.0.2
- ember-bootstrap v0.5.1
- ember-cli v1.13.13
- ember-cli-app-version v1.0.0
- ember-cli-babel v5.1.6
- ember-cli-content-security-policy v0.4.0
- ember-cli-dependency-checker v1.2.0
- ember-cli-htmlbars v1.0.2
- ember-cli-htmlbars-inline-precompile v0.3.1
- ember-cli-ic-ajax v0.2.1
- ember-cli-inject-live-reload v1.4.0
- ember-cli-jquery-ui v0.0.20
- ember-cli-qunit v1.2.1
- ember-cli-release v0.2.8
- ember-cli-shims v0.0.6
- ember-cli-sri v1.2.1
- ember-cli-test-loader v0.2.1
- ember-cli-uglify v1.2.0
- ember-d3 v0.1.0
- ember-data v2.1.0
- ember-disable-proxy-controllers v1.0.1
- ember-export-application-global v1.0.5
- ember-load-initializers v0.1.7
- ember-qunit v0.4.16
- ember-qunit-notifications v0.1.0
- ember-resolver v2.0.3
- ember-spin-spinner v0.2.3
- ember-truth helpers v1.2.0
- jquery v2.1.4
- jquery-ui v1.11.4
- loader.js v3.3.0
- momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied. See the License for the specific language governing permissions and limitations under the License.
For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT
LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY,
 WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js
--------------------------------------------------------------------------------
The MIT License
Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN 
THE SOFTWARE.

For: 
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js 
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js 
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery 
Apache HBase - Server which contains JQuery minified javascript library version 1.8.3 
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7
--------------------------------------------------------------------------------
MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of 
this software and associated documentation files (the "Software"), to deal in 
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For: 
oj! Algorithms - version 43.0
--------------------------------------------------------------------------------

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:

hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

--------------------------------------------------------------------------------

D3 is available under a 3-clause BSD license. For details, see:
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:

HSQldb Database 2.3.4

--------------------------------------------------------------------------------

(HSQL License)

"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPersonic SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity
that creates or contributes to the creation of
Modifications.

1.2. Contributor Version means the combination of the
Original Software, prior Modifications used by a Contributor (if any), and the
Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the
combination of files containing Original Software with files containing
Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other
than Source Code.

1.5. Initial Developer means the individual or entity
that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or
portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means
having the right to grant, to the maximum extent possible, whether at the time
of the initial grant or subsequently acquired, any and all of the rights
conveyed herein.

1.9. Modifications means the Source Code and Executable
form of any of the following:
A. Any file that results from an addition to,
deletion from or modification of the contents of a file containing Original
Software or previous Modifications;
B. Any new file that contains any part of the Original Software
or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work;

and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or
(ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent
Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.
3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1.
New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial
Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48 C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1
1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.
2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a
different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS
LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0
------------------------------------------------------------------------------------------------------------------
This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.
By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.
"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively “Losses”) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties
related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

-------------------------------------------------------------------------------
(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  * Neither the name of the <organization> nor the
names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0

dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective
Works, and to reproduce the Work as incorporated in the Collective Works; to create and reproduce Derivative Works; to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works; to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works. For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove
from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author").

Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1
--------------------------------------------------------------------------------
(MPL 2.0)

Mozilla Public License

Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.
1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.
1.12. Secondary License means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form means the form of the work preferred for making modifications.

1.14. You (or Your) means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or
for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:
such Covered Software must also be made available in Source Code Form, as
described in Section 3.1, and You must inform recipients of the Executable Form
how they can obtain a copy of such Source Code Form by reasonable means in a
timely manner, at a charge no more than the cost of distribution to the
recipient; and

You may distribute such Executable Form under the terms of this License, or
sublicense it under different terms, provided that the license for the
Executable Form does not attempt to limit or alter the recipients rights in
the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided
that You also comply with the requirements of this License for the Covered
Software. If the Larger Work is a combination of Covered Software with a work
governed by one or more Secondary Licenses, and the Covered Software is not
Incompatible With Secondary Licenses, this License permits You to additionally
distribute such Covered Software under the terms of such Secondary License(s),
so that the recipient of the Larger Work may, at their option, further
distribute the Covered Software under the terms of either this License or such
Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including
copyright notices, patent notices, disclaimers of warranty, or limitations of
liability) contained within the Source Code Form of the Covered Software, except
that You may alter any license notices to the extent required to remedy known
factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity
or liability obligations to one or more recipients of Covered Software. However,
You may do so only on Your own behalf, and not on behalf of any Contributor. You
must make it absolutely clear that any such warranty, support, indemnity, or
liability obligation is offered by You alone, and You hereby agree to indemnify
every Contributor for any liability incurred by such Contributor as a result of
warranty, support, indemnity or liability terms You offer. You may include
additional disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with
respect to some or all of the Covered Software due to statute, judicial order,
or regulation then You must: (a) comply with the terms of this License to the
maximum extent possible; and (b) describe the limitations and the code they
affect. Such description must be placed in a text file included with all
distributions of the Covered Software under this License. Except to the extent
prohibited by statute or regulation, such description must be sufficiently
detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You
fail to comply with any of its terms. However, if You become compliant, then the
rights granted under this License from a particular Contributor are reinstated
(a) provisionally, unless and until such Contributor explicitly and finally
terminates Your grants, and (b) on an ongoing basis, if such Contributor fails
to notify You of the non-compliance by some reasonable means prior to 60 days
after You have come back into compliance. Moreover, Your grants from a
particular Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the first
time You have received notice of non-compliance with this License from such
Contributor, and You become compliant prior to 30 days after Your receipt of the
notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions, counter-claims, and
cross-claims) alleging that a Contributor Version directly or indirectly
infringes any patent, then the rights granted to You by any and all Contributors
for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user
license agreements (excluding distributors and resellers) which have been
validly granted by You or Your distributors under this License prior to
termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without
warranty of any kind, either expressed, implied, or statutory, including,
without limitation, warranties that the Covered Software is free of defects,
merchantable, fit for a particular purpose or non-infringing. The entire risk as
to the quality and performance of the Covered Software is with You. Should any
Covered Software prove defective in any respect, You (not any Contributor)
assume the cost of any necessary servicing, repair, or correction. This
disclaimer of warranty constitutes an essential part of this License. No use of
any Covered Software is authorized under this License except under this
disclaimer.

7. Limitation of Liability
Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a
new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

/*--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

*/
3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported
License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or
other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of
fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly
granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these
credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections
Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law
includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.327 libsoup 2.52.2-1ubuntu0.3

1.327.1 Available under license:

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.
Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain
that everyone understands that there is no warranty for this free
library. If the library is modified by someone else and passed on, we
want its recipients to know that what they have is not the original
version, so that any problems introduced by others will not reflect on
the original authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that companies distributing free
software will individually obtain patent licenses, thus in effect
transforming the program into proprietary software. To prevent this,
we have made it clear that any patent must be licensed for everyone's
free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary
GNU General Public License, which was designed for utility programs. This
license, the GNU Library General Public License, applies to certain
designated libraries. This license is quite different from the ordinary
one; be sure to read it in full, and don't assume that anything in it is
the same as in the ordinary license.

The reason we have a separate public license for some libraries is that
they blur the distinction we usually make between modifying or adding to a
program and simply using it. Linking a program with a library, without
changing the library, is in some sense simply using the library, and is
analogous to running a utility program or application program. However, in
a textual and legal sense, the linked executable is a combined work, a
derivative of the original library, and the ordinary General Public License
treats it as such.

Because of this blurred distinction, using the ordinary General
Public License for libraries did not effectively promote software
sharing, because most developers did not use the libraries. We
concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the
users of those programs of all benefit from the free status of the
libraries themselves. This Library General Public License is intended to
permit developers of non-free programs to use free libraries, while
preserving your freedom as a user of such programs to change the free
libraries that are incorporated in them. (We have not seen how to achieve
this as regards changes in header files, but we have achieved it as regards
changes in the actual functions of the Library.) The hope is that this
will lead to faster development of free libraries.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License: they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the
library”. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

### 1.328 python-setuptools 28.8.0

#### 1.328.1 Available under license:

Copyright (C) 2016 Jason R Coombs <jaraco@jaraco.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.329 junit-platform-launcher 1.7.0

#### 1.329.1 Available under license:

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

### 1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
* **i)** changes to the Program, and
**ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular
Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself
or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that
are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale
of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable),
including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the
Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a
whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from,
or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form
that contains any contents of the Program. Modified Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or
subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer
of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software
source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license,
including any exceptions or additional permissions as identified by the initial Contributor.

### 2. Grant of Rights

**a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive,
worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly
perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

**b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive,
worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise
transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to
the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor,
such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license
shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

If a Contributor Distributes the Program in any form, then:

* the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
  * effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
  * effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
  * does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
  * requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

When the Program is Distributed as Source Code:

* it must be made available under this Agreement, or if the Program is combined with other material in a separate file or files made available under a Secondary License, and the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

### 5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

### 6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### 7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

#### Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the
1.330 is-bigint 1.0.1
1.330.1 Available under license:
MIT License

Copyright (c) 2018 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.331 pluralize 1.2.1
1.331.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.332 react-json-view 1.19.1 1.19.1
1.332.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Mac Gainor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.333 has-values 1.0.0 1.0.0
1.333.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.334 interpret 1.4.0 1.4.0
1.334.1 Available under license:
Copyright (c) 2014-2018 Tyler Kellen <tyler@sleekcode.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.335 t-digest 3.2
1.335.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to Ted Dunning under one or more
 * contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/FloatHistogram.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/AVLTreeDigest.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/MergingDigest.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/IntAVLTree.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/AVLGroupTree.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/Simple64.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/TDigest.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/Histogram.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/Sort.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/AbstractTDigest.java
* /opt/cola/permits/1000864524_1646171528.06/0/t-digest-3-2-sources-
jar/com/tdunning/math/stats/Centroid.java

**1.336 chalk 2.4.1**

**1.336.1 Available under license**

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.337 summernote 0.8.15 0.8.15
1.337.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.338 avatica-metrics 1.15.0
1.338.1 Available under license:

Apache Calcite Avatica Metrics
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.339 append-transform 2.0.0

1.339.1 Available under license:
The MIT License (MIT)

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.340 jsreport-data 2.4.0 2.4.0

1.340.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

### 1.341 escalade 3.1.1

**1.341.1 Available under license:**
MIT License

Copyright (c) Luke Edwards <luke.edwards05@gmail.com> (lukeed.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.342 chalk 4.1.2

**1.342.1 Available under license:**
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
1.343 css-what 2.1.3

1.343.1 Available under license:
Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.344 react-redux 5.0.7

1.344.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.345 swagger-annotations 2.0.7

1.345.1 Available under license:

```java
/**
 * Copyright 2017 SmartBear Software
 *<p>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *<p>
 * http://www.apache.org/licenses/LICENSE-2.0
 *<p>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

class License {
    public interface License {
        /**
         * The license name used for the API.
         */
    }
}
* @return the name of the license
**/
String name() default "";

/**
 * A URL to the license used for the API. MUST be in the format of a URL.
 * @return the URL of the license
 **/
String url() default "";

/**
 * The list of optional extensions
 * @return an optional array of extensions
 */
Extension[] extensions() default {};

1.346 commons-lang3 2.6
1.346.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Lang
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.347 batik-css 1.13
1.347.1 Available under license :
No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/dom/CSSOMSVGStyleDeclaration.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/parser/DefaultPseudoElementSelector.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/RGBColorValue.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/sac/AbstractAttributeCondition.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/ShorthandManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/CSSNavigableNode.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/Messages.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/svg/ColorRenderingManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/StyleMap.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/css2/FontFamilyManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/css2/FontStyleManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/css2/FontWeightManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/css2/TextDecorationManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/svg/ClipRuleManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/svg/KerningManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/parser/DefaultChildSelector.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/parser/DefaultAttributeCondition.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/SVGCSSEngine.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-jar/org/apache/batik/css/engine/value/svg/KerningManager.java
* /opt/cola/permits/1135872673_1613620733.01/0/batik-css-1-13-sources-1-
1.348 find-up 2.1.0

1.348.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.349 core 3.1.1

1.349.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0'

Found license 'Eclipse Public License 1.0' in 'Eclipse Public License Version 1.0 (quot;EPLquot;). A copy of the EPL is available'

Found license 'Eclipse Public License 1.0' in 'Eclipse Public License Version 1.0 (quot;EPLquot;). A copy of the EPL is available at &lt;a href=quot;http://www.eclipse.org/legal/epl-v10.htmlquot;&gt;http://www.eclipse.org/legal/epl-v10.html&lt;/a&gt;.'

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 &quot; * All Rights Reserved.&quot; + //NON-NLS-1S

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 compiler.copyright = Copyright IBM Corp 2000, 2006. All rights reserved.'

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 which accompanies'

1.350 base64-arraybuffer 0.1.5 0.1.5

1.350.1 Available under license:

Copyright (c) 2012 Niklas von Hertzen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.351 stream-shift 1.0.1 1.0.1

1.351.1 Available under license :
The MIT License (MIT)

Copyright (c) 2016 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.352 jackson-dataformat-hocon 1.1.1

1.352.1 Available under license :
Apache-2.0

1.353 jackson 1.0.0

1.353.1 Available under license :
luminis OSGi bundle

This product includes software developed at
luminis (http://opensource.luminis.net/).
Copyright (c) 2006, 2007 luminis
Licensed under the new BSD.

This product includes software developed at
The OSGi Alliance (http://www.osgi.org/).
Copyright (c) 2006 The OSGi Alliance
Licensed under the Apache Software License version 2.0.

This product includes software developed at
aQute SARL (http://www.aquote.biz/).
Copyright (c) 2006 aQute SARL
Licensed under the Apache Software License version 2.0.

/*
Std: LICENSE.txt,v 1.1.1.1 2004/07/01 13:59:13 jvanzy1 Exp $

Copyright 2002 (C) The Codehaus. All Rights Reserved.

Redistribution and use of this software and associated documentation
("Software"), with or without modification, are permitted provided
that the following conditions are met:

1. Redistributions of source code must retain copyright
   statements and notices. Redistributions must also contain a
   copy of this document.

2. Redistributions in binary form must reproduce the
   above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other
   materials provided with the distribution.

3. The name "classworlds" must not be used to endorse or promote
   products derived from this Software without prior written
   permission of The Codehaus. For written permission, please
   contact bob@codehaus.org.

4. Products derived from this Software may not be called "classworlds"
   nor may "classworlds" appear in their names without prior written
   permission of The Codehaus. "classworlds" is a registered
   trademark of The Codehaus.

5. Due credit should be given to The Codehaus.
   (http://classworlds.codehaus.org/).

THIS SOFTWARE IS PROVIDED BY THE CODEHAUS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
THE CODEHAUS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

*/
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000,2002,2003 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
 attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This copy of Jackson JSON processor is licensed under the
Lesser General Public License (LGPL), version 2.1 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.gnu.org/licenses/licenses.html

A copy is also included with both the the downloadable source code package
and jar that contains class bytecodes, as file "LGPL 2.1". In both cases,
that file should be located next to this file: in source distribution
the location should be "release-notes/lgpl"; and in jar "META-INF/

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

=========================================================================
== NOTICE file corresponding to the section 4 d of
== the Apache License, Version 2.0, in this case for the Apache Ant distribution. ==
=========================================================================
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

This product includes also software developed by:
- the W3C consortium (http://www.w3c.org),
- the SAX project (http://www.saxproject.org)

Please read the different LICENSE files present in the root directory of this distribution.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (c) 2006, Javolution (http://javolution.org)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
    Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/
*
* TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
*
* 1. Definitions.
*
* "License" shall mean the terms and conditions for use, reproduction,
* and distribution as defined by Sections 1 through 9 of this document.
*
* "Licensors" shall mean the copyright owner or entity authorized by
* the copyright owner that is granting the License.
*
* "Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common 
control with that entity. For the purposes of this definition, 
"control" means (i) the power, direct or indirect, to cause the 
direction or management of such entity, whether by contract or 
otherwise, or (ii) ownership of fifty percent (50%) or more of the 
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity 
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, 
including but not limited to software source code, documentation 
source, and configuration files.

"Object" form shall mean any form resulting from mechanical 
transformation or translation of a Source form, including but 
not limited to compiled object code, generated documentation, 
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or 
Object form, made available under the License, as indicated by a 
copyright notice that is included in or attached to the work 
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object 
form, that is based on (or derived from) the Work and for which the 
editorial revisions, annotations, elaborations, or other modifications 
represent, as a whole, an original work of authorship. For the purposes 
of this License, Derivative Works shall not include works that remain 
separable from, or merely link (or bind by name) to the interfaces of, 
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including 
the original version of the Work and any modifications or additions 
to that Work or Derivative Works thereof, that is intentionally 
submitted to Licensor for inclusion in the Work by the copyright owner 
or by an individual or Legal Entity authorized to submit on behalf of 
the copyright owner. For the purposes of this definition, "submitted" 
means any form of electronic, verbal, or written communication sent 
to the Licensor or its representatives, including but not limited to 
communication on electronic mailing lists, source code control systems, 
and issue tracking systems that are managed by, or on behalf of, the 
Licensor for the purpose of discussing and improving the Work, but 
excluding communication that is conspicuously marked or otherwise 
designated in writing by the copyright owner as "Not a Contribution.”

"Contributor" shall mean Licensor and any individual or Legal Entity 
on behalf of whom a Contribution has been received by Licensor and
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed...
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

* 9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

* END OF TERMS AND CONDITIONS

* APPENDIX: How to apply the Apache License to your work.

* To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

* Copyright [yyyy] [name of copyright owner]

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

This copy of Jackson JSON processor is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.
You may obtain a copy of the License at:

http://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package
and jar that contains class bytecodes, as file "ASL 2.0". In both cases,
that file should be located next to this file: in source distribution
the location should be "release-notes/asl"; and in jar "META-INF/"

/*************************************************************************
* CruiseControl, a Continuous Integration Toolkit
* Copyright (c) 2001-2003, ThoughtWorks, Inc.
* 651 W Washington Ave. Suite 500
* Chicago, IL 60661 USA
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* + Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* + Redistributions in binary form must reproduce the above
* copyright notice, this list of conditions and the following
* disclaimer in the documentation and/or other materials provided
* with the distribution.
* + Neither the name of ThoughtWorks, Inc., CruiseControl, nor the
* names of its contributors may be used to endorse or promote
* products derived from this software without specific prior
* written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*************************************************************************/

// NOTICE file corresponding to the section 4d of The Apache License,
Maven Wagon API
Copyright 2003-2006 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Maven Ant Task
Copyright 2002-2008 The Apache Software Foundation

This product includes/uses software(s) developed by ‘an unknown organization’
   - Unnamed - ant:ant:jar:1.6.2

This product includes/uses software(s) developed by 'Apache Software Foundation' (http://www.apache.org/)
   - Maven Wagon File Provider (http://maven.apache.org/wagon/wagon-providers/wagon-file)
   - Maven Wagon API (http://maven.apache.org/wagon/wagon-provider-api)
   - XML Commons External Components XML APIs (http://xml.apache.org/commons/#external)

This product includes/uses software(s) developed by 'Codehaus' (http://www.codehaus.org/)
   - Default Plexus Container
   - Plexus Common Utilities (http://plexus.codehaus.org/plexus-utils)

This product includes/uses software(s) developed by 'JTidy' (http://jtidy.sf.net)
   - JTidy (http://jtidy.sourceforge.net)

This product includes/uses software(s) developed by 'The Apache Software Foundation' (http://www.apache.org/)
   - Maven Artifact (http://maven.apache.org/maven-artifact)
   - Maven Artifact Manager (http://maven.apache.org/maven-artifact-manager)
   - Maven Error Diagnostics (http://maven.apache.org/maven-error-diagnostics)
   - Maven Model (http://maven.apache.org/maven-model)
   - Maven Plugin Registry Model (http://maven.apache.org/maven-plugin-registry)
   - Maven Profile Model (http://maven.apache.org/maven-profile)
   - Maven Project Builder (http://maven.apache.org/maven-project)
   - Maven Repository Metadata Model (http://maven.apache.org/maven-repository-metadata)
   - Maven Local Settings Model (http://maven.apache.org/maven-settings)

This product includes/uses software(s) developed by 'The Codehaus' (http://codehaus.org/)
   - classworlds (http://classworlds.codehaus.org/)
This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.

For additional credits (generally to people who reported problems) see CREDITS file.

/* ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000-2002 The Apache Software Foundation. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 *
 * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 * "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED `AS IS' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF...
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND 
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, 
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT 
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF 
* SUCH DAMAGE.
* ====================================================================
* 
* This software consists of voluntary contributions made by many 
* individuals on behalf of the Apache Software Foundation. For more 
* information on the Apache Software Foundation, please see 
* */

Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions 
are met:

1. Redistributions of source code must retain the above copyright notice, 
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright 
   notice, this list of conditions and the following disclaimer in 
   the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, 
   must include the following acknowledgment:

   "This product includes software developed by the Indiana University 
   Extreme! Lab (http://www.extreme.indiana.edu/)."

   Alternately, this acknowledgment may appear in the software itself, 
   if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana University" and "Indiana University Extreme! Lab" 
   must not be used to endorse or promote products derived from this 
   software without prior written permission. For written permission, 
   please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University" 
   name nor may "Indiana University" appear in their name, without prior 
   written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.354 commons-lang3 3.11

1.354.1 Available under license:

Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.355 zlib 1.2.1

1.355.1 Available under license:

/* zlib.h -- interface of the 'zlib' general purpose compression library
 version 1.2.1, November 17th, 2003

Copyright (C) 1995-2003 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu
The data format used by the zlib library is described by RFCs (Request for
Comments) 1950 to 1952 in the files http://www.ietf.org/rfc/rfc1950.txt
(zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format).

1.356 jackson-dataformat-yaml 2.9.9

1.356.1 Available under license :

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.
This copy of Jackson JSON processor YAML module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.357 for-in 1.0.2 1.0.2

1.357.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.358 nan 1.3.0 1.3.0

1.358.1 Available under license:

Copyright 2013, NAN contributors:
- Rod Vagg <https://github.com/rvagg>
- Benjamin Byholm <https://github.com/kkoopa>
- Trevor Norris <https://github.com/trevnorris>
- Nathan Rajlich <https://github.com/TooTallNate>
- Brett Lawson <https://github.com/brett19>
- Ben Noordhuis <https://github.com/bnoordhuis>
(the "Original Author")
All rights reserved.

MIT +no-false-attribs License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Distributions of all or part of the Software intended to be used by the recipients as they would use the unmodified Software,
containing modifications that substantially alter, remove, or
disable functionality of the Software, outside of the documented
configuration mechanisms provided by the Software, shall be
modified such that the Original Author's bug reporting email
addresses and urls are either replaced with the contact information
of the parties responsible for the changes, or removed entirely.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Except where noted, this license applies to any and all software
programs and associated documentation files created by the
Original Author, when distributed with the Software.

1.359 react-textarea-autosize 6.1.0 6.1.0

1.359.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 Andrey Popp

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.360 uri-js 4.4.0 4.4.0
1.360.1 Available under license:

Copyright 2011 Gary Court. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY GARY COURT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL GARY COURT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of Gary Court.

1.361 call-bind 1.0.2 1.0.2
1.361.1 Available under license:

MIT License

Copyright (c) 2020 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.362 configstore 5.0.1
1.362.1 Available under license :
BSD 2-Clause License

Copyright (c) Google
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.363 msgpack 0.6.6
1.363.1 Available under license :
No license file was found, but licenses were detected in source scan.

// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,
Found in path(s):
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/MessagePackable.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/AnyTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/packer/AbstractPacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/builder/TemplateBuilderChain.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/type/ValueFactory.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/util/json/JSONPacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/MessagePackableTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/DoubleArrayTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/packer/Unconverter.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/annotation/Message.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/IntegerArrayTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/io/BufferedOutput.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/type/AbstractBooleanValue.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/packer/PackerStack.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-jar/org/msgpack/io/StreamOutput.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/type/StringRawValueImpl.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/builder/BeansBuildContext.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/annotation/OrdinalEnum.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/annotation/Optional.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/template/ShortArrayTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/io/EndOfBufferException.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/unpacker/BufferUnpacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1jar/org/msgpack/type/BooleanValue.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-

Open Source Used In DNAC Platform Arica 1345


Open Source Used in DNAC Platform

Arica 1347

* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/DoubleTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/type/ValueType.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/DefaultFieldEntry.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/type/DoubleValueImpl.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/UnpackerIterator.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/util/json/JSONUnpacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/Converter.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/io/AbstractInput.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/BigIntegerTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/ShortTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/ValueTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/ReflectionBeansTemplateBuilder.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/AbstractUnpacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/MessageTypeException.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/util/TemplatePrecompiler.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/packer/BufferPacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/TemplateReference.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/ValueAccept.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/AbstractTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/annotation/Index.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/TemplateBuilder.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/AbstractTemplateBuilder.java
Open Source Used In DNAC Platform Artica 1348

* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/FieldOption.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/io/ByteBufferOutput.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/ReflectionTemplateBuilder.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/LongTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/packer/Packer.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/JavaassistTemplateBuilder.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/Template.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/LongAccept.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/Templates.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/GenericCollectionTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/annotation/MessagePackBeans.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/annotation/NotNullable.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/packer/MessagePackPacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/type/TrueValueImpl.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/ByteTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/DoubleAccept.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/io/LinkedBufferInput.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/io/BufferReferer.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/MessagePackUnpacker.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/annotation/Delegate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/annotation/Ignore.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/unpacker/UnpackerStack.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/BooleanTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/StringTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1.jar/org/msgpack/template/builder/TemplateBuildException.java
jar/org/msgpack/template/ListTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-
jar/org/msgpack/template/CharacterTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-
jar/org/msgpack/template/IntegerTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-
jar/org/msgpack/template/LongArrayTemplate.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-jar/org/msgpack/io/Output.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-jar/org/msgpack/type/ArrayValue.java
* /opt/cola/permits/1138552721_1614291037.54/0/msgpack-0-6-6-sources-1-jar/org/msgpack/template/BigDecimalTemplate.java

1.364 color-name 1.1.4
1.364.1 Available under license:
The MIT License (MIT)
Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.365 jpam 1.1
1.365.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
* Copyright 2003-2007 Greg Luck

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1126356109_1611624421.8/0/jpam-1-1-src-zip/JPam-1.1/java/net/sf/jpam/Pam.java
* /opt/cola/permits/1126356109_1611624421.8/0/jpam-1-1-src-zip/JPam-1.1/java/net/sf/jpam/jaas/JpamLoginModule.java
* /opt/cola/permits/1126356109_1611624421.8/0/jpam-1-1-src-zip/JPam-1.1/java/net/sf/jpam/PamReturnValue.java
* /opt/cola/permits/1126356109_1611624421.8/0/jpam-1-1-src-zip/JPam-1.1/java/net/sf/jpam/PamException.java
No license file was found, but licenses were detected in source scan.

License: Apache Software License, v. 1.1

Found in path(s):
* /opt/cola/permits/1126356109_1611624421.8/0/jpam-1-1-src-zip/JPam-1.1/rpm/jpam.spec

1.366 hbase-shaded-gson 3.4.1

1.366.1 Available under license:

Apache HBase Relocated (Shaded) GSON Libs
Copyright 2017-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.367 glib 2.48.2 0ubuntu4.4

1.367.1 Available under license:

This work may be reproduced and distributed in whole or in part, in any medium, physical or electronic, so as long as this copyright notice remains intact and unchanged on all copies. Commercial redistribution is permitted and encouraged, but you may not redistribute, in whole or in part, under terms more restrictive than
BECAUSE THIS WORK IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE WORK, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE WORK "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. SHOULD THE WORK PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE WORK AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE WORK, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.
This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.
GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion
of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices
stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no
charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a
table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility
is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or
table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.
3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.
If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
That's all there is to it!

1.368 ical4j 1.0.2

1.368.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2011, Ben Fortuna
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * o Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * o Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 *
 * o Neither the name of Ben Fortuna nor the names of any other contributors
 * may be used to endorse or promote products derived from this software
 * without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
 * A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
 * PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
 * LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
 * NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/TzOffsetToTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/data/CalendarBuilderCustomRegistryTest.java
jar/net/fortuna/ical4j/model/property/StatusTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/parameter/FbTypeTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/parameter/ValueTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/PriorityTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/DateTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/RDateTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/CompletedTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/EncodingTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/TypeTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/RelatedToTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-
jar/net/fortuna/ical4j/model/property/PostalcodeTest.java

Open Source Used In DNAC Platform Arica

jar/net/fortuna/ical4j/model/property/ProdIdTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/util/CalendarsTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/data/CalendarParserFactoryTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/ResourcesTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/data/CalendarParserImpITest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/TranspTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/VAvailabilityTestCase.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/DelegatedToTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/UrlTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/XComponentTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/RegionTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/PercentCompleteTest.java
* /opt/cola/permits/1000715208_1614362208.0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/LocationTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/ClazzTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/FreeBusyTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/VJournalTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/StreetAddressTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/VAlarmTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/StandardTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/TzOffsetFromTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/RequestStatusTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/VvenueTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/ActionTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/TriggerTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/DateTimeTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/parameter/FmtTypeTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/parameter/UidGeneratorTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/DtStartTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/TextListTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/IndexedComponentListTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/SentByTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/LocationTypeTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/TzIdTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/CalendarTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/property/ExDateTest.java
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/net/fortuna/ical4j/model/component/VAvailabilityTest.java
No license file was found, but licenses were detected in source scan.

# All rights reserved.
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
#   o Redistributions of source code must retain the above copyright
#     notice, this list of conditions and the following disclaimer.
#   o Redistributions in binary form must reproduce the above copyright
#     notice, this list of conditions and the following disclaimer in the
# documentation and/or other materials provided with the distribution.
# o Neither the name of Ben Fortuna nor the names of any other contributors
# without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/log4j.properties
* /opt/cola/permits/1000715208_1614362208.0/0/ical4j-1-0-2-test-sources-jar/ical4j.properties

1.369 is-extendable 0.1.1 0.1.1

1.369.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.370 h2 1.4.200

1.370.1 Available under license :

Mozilla Public License
Version 2.0
1. Definitions
1.1. “Contributor”
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. “Contributor Version”
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s Contribution.
1.3. “Contribution”
means Covered Software of a particular Contributor.

1.4. “Covered Software”
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form
of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. “Incompatible With Secondary Licenses”
means
that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also
under the terms of a Secondary License.

1.6. “Executable Form”
means any form of the work other than Source Code Form.

1.7. “Larger Work”
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered
Software.

1.8. “License”
means this document.

1.9. “Licensable”
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or
subsequently, any and all of the rights conveyed by this License.

1.10. “Modifications”
means any of the following:
any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of
Covered Software; or
any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent
Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using,
selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. “Secondary License”
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1,
the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form”
means the form of the work preferred for making modifications.
1.14. “You” (or “Your”) means an individual or a legal entity exercising rights under this License. For legal entities, “You” includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants
Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date
The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope
The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses
No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation
Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use
This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions
Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities
3.1. Distribution of Source Form
All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients’ rights in the Source Code Form.

3.2. Distribution of Executable Form
If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients’ rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work
You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices
You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms
You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the...
maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination
5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty
Covered Software is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability
Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation
Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.
9. Miscellaneous
This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License
10.1. New Versions
Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions
You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions
If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses
If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - “Incompatible With Secondary Licenses” Notice
This Source Code Form is “Incompatible With Secondary Licenses”, as defined by the Mozilla Public License, v. 2.0.
H2 is dual licensed and available under the MPL 2.0 (Mozilla Public License Version 2.0) or under the EPL 1.0 (Eclipse Public License).

1.371 commons-compress 1.2.2 1.2.2
1.371.1 Available under license:
Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.372 prompt-tmp 1.0.0 1.0.0

1.372.1 Available under license:

Copyright (c) 2010 Nodejitsu Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.373 cypress 1.0.0 1.0.0
1.373.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Cypress.io, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

import cs from 'classnames'
import React, { Component } from 'react'
import { observer } from 'mobx-react'

import ipc from '../lib/ipc'
import errors from '../lib/errors'
import projectsStore from '../projects/projects-store'

const SUCCESS = 'SUCCESS'
const FAILURE = 'FAILURE'

@observer
class PermissionMessage extends Component {
    constructor (props) {
        super(props)

        this.state = {
            error: null,
            isSubmitting: false,
            result: null,
        }
    }

    render () {
        return (
            <div
                className={cs('request-access', {
                    'is-submitting': this.state.isSubmitting,
                })}
                
            
        )
    }
}
function _message() {
    const membershipRequested = projectsStore.wasMembershipRequested(this.props.project.id)

    if (this.state.result === SUCCESS || membershipRequested) {
        return this._success()
    } else if (this.state.result === FAILURE) {
        return this._failure()
    } else {
        return this._noResult()
    }
}

function _button() {
    return (
        <button
            className='btn btn-primary'
            disabled={this.state.isSubmitting}
            onClick={this._requestAccess}
        >
            <span>
                <i className='fa fa-paper-plane'></i>{' '}
                Request Access
            </span>
            <i className='fa fa-spinner fa-spin'></i>
        </button>
    )
}

function _success() {
    return (
        <div className='empty'>
            <h4>
                <i className='fa fa-check passed'></i>{' '}
                Request Sent
            </h4>
            <p>
                The project owner will be notified with your request.
            </p>
        </div>
    )
}
_failure ()
const error = this.state.error

// if they're denied or the request has already been made,
// tell them it's all good
if (errors.isDenied(error) || errors.isAlreadyRequested(error)) {
  return this._success()
} else {
  return (
    <div className='empty'>
      <h4>
        <i className='fa fa-exclamation-triangle fa-li'></i> Request Failed
      </h4>
      <p>An unexpected error occurred while requesting access:</p>
      <pre className='alert alert-danger'>{this.state.error.message}</pre>
      <p>Try again.</p>
      {this._button()}
    </div>
  )
}

_noResult ()
return (
  <div className="empty">
    <h4>
      <i className='fa fa-lock fa-li'></i> Request access to see the runs
    </h4>
    <p>This is a private project created by someone else.</p>
    <p>The project owner must give you access to see the runs.</p>
    {this._button()}
  </div>
)

_requestAccess = () =&gt; {
  this.setState({
    isSubmitting: true,
  })

  const id = this.props.project.id

  ipc.requestAccess(id)
  .then(() =&gt; {

})
projectsStore.membershipRequested(id)
this._ setResult()
})
catch(ipc.isUnauthed, ipc.handleUnauthed)
catch((error) => {
    this._ setResult(error)
})

_setResult (error) {
    if (errors.isAlreadyMember(error)) {
        this.props.onRetry()
        return
    }

    this.setState({
        error,
        isSubmitting: false,
        result: error ? FAILURE : SUCCESS,
    })
}

export default PermissionMessage

1.374 jackson-annotations 2.12.3
1.374.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
This License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.375 color-convert 2.0.1 2.0.1

1.375.1 Available under license:

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.376 zip-stream 2.1.3 2.1.3

1.376.1 Available under license:

Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.377 istanbul-lib-instrument 4.0.3
1.377.1 Available under license:
Copyright 2012-2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.378 @babel/plugin-transform-unicode-regex 7.16.0

1.378.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.379 jooq 3.13.1

1.379.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.380 nconf 0.10.0 0.10.0

1.380.1 Available under license:
Copyright (C) 2011 Charlie Robbins and the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.381 swagger-core 1.5.0

1.382 deep-is 0.1.3

1.382.1 Available under license:
MIT
1.383 estraverse 5.2.0

1.383.1 Available under license:
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.384 drill-jdbc-storage 1.19.0

1.384.1 Available under license:

Drill : Contrib : Storage : JDBC
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.385 swagger-jersey2-jaxrs 1.5.0

1.386 extend 3.0.2 3.0.2

1.386.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Stefan Thomas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.387 jsreport-pdfjs 0.5.1 0.5.1

1.387.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013-2017 Markus Ast

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.388 express 4.17.1 4.17.1

1.388.1 Available under license:
(The MIT License)

Copyright (c) 2009-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2013-2014 Roman Shtylman <shtylman+expressjs@gmail.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.389 redux-act 1.7.4 1.7.4

1.389.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, 
unless required by applicable law (such as deliberate and grossly 
negligent acts) or agreed to in writing, shall any Contributor be 
liable to You for damages, including any direct, indirect, special, 
incidental, or consequential damages of any character arising as a 
result of this License or out of the use or inability to use the 
Work (including but not limited to damages for loss of goodwill, 
work stoppage, computer failure or malfunction, or any and all 
other commercial damages or losses), even if such Contributor 
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing 
the Work or Derivative Works thereof, You may choose to offer, 
and charge a fee for, acceptance of support, warranty, indemnity, 
or other liability obligations and/or rights consistent with this 
License. However, in accepting such obligations, You may act only 
on Your own behalf and on Your sole responsibility, not on behalf 
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.390 braces 1.8.5

1.390.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in 
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.391 sparse-bitfield 3.0.3 3.0.3
1.391.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.392 babel-helpers 6.24.1 6.24.1
1.392.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.393 joda-time 2.9.1

1.393.1 Available under license:

=============================================================================  
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
=============================================================================  
This product includes software developed by  
Joda.org (http://www.joda.org/).

Apache License  
Version 2.0, January 2004  
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.394 parquet-column 1.8.1
1.394.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

--------------------------------------------------------------------------------
This product depends on fastutil and includes it in this binary artifact.
Fastutil provides type-specific collection implementations.

Copyright: 2002-2014 Sebastiano Vigna
Home page: http://fasutil.di.unimi.it/
License: http://www.apache.org/licenses/LICENSE-2.0.html

Apache Parquet Column
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.395 levn 0.3.0
1.395.1 Available under license :
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.396 keyv 3.1.0

1.396.1 Available under license:
MIT License

Copyright (c) 2017 Luke Childs

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.397 ms 2.1.2 2.1.2

1.397.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.398 groovy-json 3.0.8

1.398.1 Available under license:

No license file was found, but licenses were detected in source scan.

# Licensed to the Apache Software Foundation (ASF) under one
# or more contributor license agreements. See the NOTICE file
# distributed with this work for additional information
# regarding copyright ownership. The ASF licenses this file
# to you under the Apache License, Version 2.0 (the
# "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing,
# software distributed under the License is distributed on an
# "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
# KIND, either express or implied. See the License for the
# specific language governing permissions and limitations
# under the License.
#
# org.apache.groovy.json.DefaultFastStringServiceFactory

Found in path(s):
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/META-INF/services/org.apache.groovy.json.FastStringServiceFactory

No license file was found, but licenses were detected in source scan.
/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/ValueMap.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/ValueMapImpl.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/Sys.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/JsonParserLax.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/JsonParserUsingCharacterSource.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/CharacterSource.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/internal/NumberValue.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-
  jar/org/apache/groovy/json/JsonTokenType.java
jar/org/apache/groovy/json/internal/CacheType.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/internal/Cache.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/groovy/json/JsonParser.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/internal/Value.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/groovy/json/StreamingJsonBuilder.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/FastStringServiceFactory.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/internal/FastStringUtils.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/groovy/json/JsonGenerator.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/internal/Exceptions.java
* /opt/cola/permits/1218308217_1634929155.36/0/groovy-json-3-0-8-sources-1-jar/org/apache/groovy/json/internal/ReaderCharacterSource.java

1.399 aws-java-sdk-kms 1.11.704

1.399.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
 * and limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 */
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
* the License. A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
* and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/RevokeGrantRequestProtocolMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyWithoutPlaintextRequestProtocolMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyWithoutPlaintextRequestProtocolMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/OriginType.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/KeyMetadata.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/TagException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyWithoutPlaintextRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyPairResultJsonUnmarshaller.java

Open Source Used In DNAC Platform Arica 1422

jar/com/amazonaws/services/kms/AbstractAWSKMS.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/GetKeyPolicyResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/DecryptRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CustomKeyStoreNotFoundExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/ExpiredImportTokenExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CreateAliasRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/InvalidCiphertextException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/CreateKeyRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/ConnectCustomKeyStoreResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/ListAliasesResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/UpdateCustomKeyStoreResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/InvalidKeyUsageExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/DependencyTimeoutExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CustomKeyStoresListEntryJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CreateGrantRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/VerifyRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/DeleteImportedKeyMaterialRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CustomKeyStoresListEntryJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/CreateAliasRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/VerifyRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
jar/com/amazonaws/services/kms/model/transform/DependencyTimeoutExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/CloudHsmClusterNotFoundException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/KMSInvalidSignatureException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/DeleteCustomKeyStoreRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/UpdateCustomKeyStoreRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/InvalidCiphertextExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/EnableKeyResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/RetireGrantRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/DisableKeyResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/CreateGrantResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ReEncryptRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/UpdateCustomKeyStoreRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ListRetirableGrantsResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/IncorrectKeyException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateDataKeyWithoutPlaintextResult.java

Open Source Used In DNAC Platform Arica 1424
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GetParametersForImportResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/GenerateRandomResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ListResourceTagsResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/KeyState.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GrantListEntryMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CustomerMasterKeySpec.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/DescribeKeyRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ListAliasesRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ExpirationModelType.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/AliasListEntryJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/DeleteCustomKeyStoreRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/WrappingKeySpec.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/DescribeKeyResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/MessageType.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/UpdateKeyDescriptionRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyWithoutPlaintextResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/PutKeyPolicyRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/KMSInvalidStateException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/AliasListEntryJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/DeleteCustomKeyStoreRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/InvalidMarkerException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CreateAliasResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/IncorrectKeyMaterialExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/CreateGrantRequestProtocolMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/ListGrantsRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/ListKeysRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/EncryptRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CreateCustomKeyStoreResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/EncryptResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/InvalidGrantIdException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/Tag.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/IncorrectTrustAnchorException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/ListKeyPoliciesResult.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CreateCustomKeyStoreResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/InvalidAliasNameExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CreateCustomKeyStoreRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CreateKeyResultJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/DisableKeyRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/CustomKeyStoreHasCMKsExceptionUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GrantConstraintsMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/AWSKMSException.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/EncryptRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/GrantConstraintsJsonUnmarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/RevokeGrantRequestMarshaller.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/SignRequest.java
* /opt/cola/permits/1137154742_1614023258.66/0/aws-java-sdk-kms-1-11-704-sources-jar/com/amazonaws/services/kms/model/transform/KMSInvalidSignatureExceptionUnmarshaller.java

Open Source Used In DNAC Platform Arica 1430
Open Source Used In DNAC Platform Arica

1.400 is-utf8 0.2.1 0.2.1

1.400.1 Available under license:

The MIT License (MIT)

Copyright (C) 2014 Wei Fanzhe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.401 color-convert 1.9.3 1.9.3

1.401.1 Available under license :

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.402 gir1.2-gdkpixbuf-2.0 2.32.2-1ubuntu1.6

1.402.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's
free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary
GNU General Public License, which was designed for utility programs. This
license, the GNU Library General Public License, applies to certain
designated libraries. This license is quite different from the ordinary
one; be sure to read it in full, and don’t assume that anything in it is
the same as in the ordinary license.

The reason we have a separate public license for some libraries is that
they blur the distinction we usually make between modifying or adding to a
program and simply using it. Linking a program with a library, without
changing the library, is in some sense simply using the library, and is
analogous to running a utility program or application program. However, in
a textual and legal sense, the linked executable is a combined work, a
derivative of the original library, and the ordinary General Public License
treats it as such.

Because of this blurred distinction, using the ordinary General
Public License for libraries did not effectively promote software
sharing, because most developers did not use the libraries. We
concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the
users of those programs of all benefit from the free status of the
libraries themselves. This Library General Public License is intended to
permit developers of non-free programs to use free libraries, while
preserving your freedom as a user of such programs to change the free
libraries that are incorporated in them. (We have not seen how to achieve
this as regards changes in header files, but we have achieved it as regards
changes in the actual functions of the Library.) The hope is that this
will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, while the latter only
works together with the library.

Note that it is possible for a library to be covered by the ordinary
General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which
contains a notice placed by the copyright holder or other authorized
party saying it may be distributed under the terms of this Library
General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.
b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)
Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception,
the source code distributed need not include anything that is normally
distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safer to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the
Free Software Foundation, Inc., 59 Temple Place - Suite 330,
Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.403 regjsgen 0.5.2
1.403.1 Available under license :
The MIT License (MIT)

Copyright 2014-2020 Benjamin Tan <https://ofcr.se/>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.404 javax-el 3.0.0
1.404.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must
make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and
supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's
software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.405 path-parse 1.0.6
1.405.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Javier Blanco

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.406 transaction-api 1.1

1.406.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
Version 1.0

* 

1. Definitions.
   o

1.1. ?Contributor? means each individual or entity that creates or contributes to the creation of Modifications.

1.2. ?Contributor Version? means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. ?Covered Software? means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. ?Executable? means the Covered Software in any form other than Source Code.

1.5. ?Initial Developer? means the individual or entity that first makes Original Software available under this License.

1.6. ?Larger Work? means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. ?License? means this document.

1.8. ?Licensable? means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. ?Modifications? means the Source Code and Executable form of any of the following:

   +

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or

+ C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. ?Original Software? means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. ?Patent Claims? means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. ?Source Code? means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. ?You? (or ?Your?) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, ?You? includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, ?control? means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

+ (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

+ (b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

+ (c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.
3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients? rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient?s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.
4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software? (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation? as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable under applicable law.
enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

*  

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

1.407 jackson-dataformat-xml 2.10.4

1.407.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits
A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.408 netty-buffer 4.0.48.Final
1.408.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnsafeHeapSwappedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnsafeDirectSwappedByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
*/
Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/WrappedUnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/SimpleLeakAwareCompositeByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/AdvancedLeakAwareCompositeByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnpooledUnsafeNoCleanerDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/WrappedCompositeByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PooledHeapByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
Found in path(s):
  */
  /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBufAllocatorMetric.java
  */
  /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBufAllocatorMetricProvider.java
  */
  /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PooledByteBufAllocatorMetric.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
  */
  /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/META-INF/maven/io.netty/netty-buffer/pom.xml

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/DuplicatedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnpooledByteBufAllocator.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/SlicedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PooledDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBufAllocator.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBufUtil.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/package-info.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBufInputStream.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/AbstractByteBufAllocator.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PooledByteBufAllocator.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnpooledHeapByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolSubpage.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PooledByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolThreadCache.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolChunkList.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolChunk.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/ByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/ByteBufOutputStream.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/PoolArena.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/ByteBufHolder.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/FixedCompositeByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/ReadOnlyUnsafeDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/AbstractDerivedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/PooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/WrappedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/AdvancedLeakAwareByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/ReadOnlyByteBufferBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/ByteBufProcessor.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/SimpleLeakAwareByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/EmptyByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
  jar/io/netty/buffer/AbstractReferenceCountedByteBuf.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolChunkMetric.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolArenaMetric.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolSubpageMetric.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/PoolChunkListMetric.java
* /opt/cola/permits/1202447202_1631568144.52/0/netty-buffer-4-0-48-final-sources-jar/io/netty/buffer/UnsafeByteBufUtil.java
1.409 qs 4.0.0 4.0.0

1.409.1 Available under license:

Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors

1.410 send 0.13.1 0.13.1

1.410.1 Available under license:

(The MIT License)

Copyright (c) 2012 TJ Holowaychuk
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.411 groovy-docgenerator 3.0.8

1.411.1 Available under license:
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE
2.0

Unless required by applicable law or agreed to in writing,
software distributed under the License is distributed on an
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License for the
specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/template.allclasses-frame.html
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/index.html
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/template.overview-frame.html
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/template.package-summary.html
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources
No license file was found, but licenses were detected in source scan.

/*! normalize.css v2.1.0 | MIT License | git.io/normalize */

Found in path(s):
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/stylesheet.css

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-jar/org/apache/groovy/docgenerator/DocGenerator.groovy

No license file was found, but licenses were detected in source scan.

# to you under the Apache License, Version 2.0 (the
# "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# software distributed under the License is distributed on an

Found in path(s):
* /opt/cola/permits/1218307070_1634931267.31/0/groovy-docgenerator-3-0-8-sources-
1.412 path-loader 1.0.10 1.0.10

1.412.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Jeremy Whitlock

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.413 type-check 0.3.2 0.3.2

1.413.1 Available under license:

Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.414 path-is-inside 1.0.2

1.414.1 Available under license:
Dual licensed under WTFPL and MIT:

---

Copyright 20132016 Domenic Denicola <d@demonic.me>

This work is free. You can redistribute it and/or modify it under the
terms of the Do What The Fuck You Want To Public License, Version 2,
as published by Sam Hocevar. See below for more details.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004

Copyright (C) 2004 Sam Hocevar <sam@hocevar.net>

Everyone is permitted to copy and distribute verbatim or modified
copies of this license document, and changing it is allowed as long
as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

---

The MIT License (MIT)

Copyright 20132016 Domenic Denicola <d@demonic.me>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.415 has-value 1.0.0 1.0.0

1.415.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.416 p-try 1.0.0

1.416.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.417 resolve-from 2.0.0 2.0.0

1.417.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.418 guava 19.0
1.419 ws 1.1.1 1.1.1

1.419.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.420 object-inspect 1.11.0

1.420.1 Available under license:

MIT License

Copyright (c) 2013 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.421 p-locate 4.1.0

1.421.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.422 slf4j-log4j 2.8.2

1.422.1 Available under license:

Apache Log4j SLF4J Binding
Copyright 1999-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.423 mime-pull 1.9.6

1.423.1 Available under license:

Found license 'General Public License 2.0' in: * Copyright (c) 1997-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in: * Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in: * Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in: * Copyright (c) 1997-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in: * Copyright (c) 1997-2015 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

1.424 hbase-shaded-netty 3.4.1

1.424.1 Available under license:

Apache HBase Relocated (Shaded) Netty Libs
Copyright 2017-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
extcept as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.425 sac 1.3

1.425.1 Available under license:

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/

This W3C work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body of any redistributed or derivative code: "Copyright [date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/

Notice of any changes or modifications to the W3C files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

____________________________________

This formulation of W3C's notice and license became active on August 14 1998 so as to improve compatibility with GPL. This version ensures that W3C software licensing terms are no more restrictive than GPL and consequently W3C software may be distributed in GPL packages. See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.
1.426 hadoop-core 1.2.1

1.426.1 Available under license:

JZlib 0.0.* were released under the GNU LGPL license. Later, we have switched over to a BSD-style license.

```
Copyright (c) 2000,2001,2002,2003 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

${project.name} is an effort undergoing incubation at the Apache Software Foundation (ASF), sponsored by the Apache Incubator PMC.

Incubation is required of all newly accepted projects until a further review indicates that the infrastructure, communications, and decision making process have stabilized in a manner consistent with other successful ASF projects.

While incubation status is not necessarily a reflection of the completeness or stability of the code, it does indicate that the project has yet to be fully endorsed by the ASF.

Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2003-2008, Objenesis Team and all contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
/* Copyright (c) 1995-2000, The Hypersonic SQL Group.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
 *
 * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 *
 * Neither the name of the Hypersonic SQL Group nor the names of its
 * contributors may be used to endorse or promote products derived from this
 * software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE HYPersonic SQL GROUP,
 * OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * This software consists of voluntary contributions made by many individuals
 * on behalf of the Hypersonic SQL Group.
 *
* For work added by the HSQL Development Group:
* Copyright (c) 2001-2004, The HSQL Development Group
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
*
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*
* Neither the name of the HSQL Development Group nor the names of its
* contributors may be used to endorse or promote products derived from this
* software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL
 * nor the names of its contributors may be used to endorse or
 * promote products derived from this software without specific prior
 * written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
 * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
 * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 */

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2000-2007, jMock.org
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of jMock nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Mockito license - MIT.

Libraries used:

Cglib - Apache License 2.0
ASM - BSD license

Mockito all distribution:

Objenesis - MIT license
Hamcrest - BSD license
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

------------------------------------------------------------------------------
License for JCraft JSch package
------------------------------------------------------------------------------

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License for jQuery

Copyright (c) 2007 John Resig, http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Thrift Software License
Copyright (c) 2006- Facebook, Inc.

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Apache Ivy

Copyright 2007-2009 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of Ivy were originally developed by
Jayasoft SARL (http://www.jayasoft.fr/)
and are licensed to the Apache Software Foundation under the
"Software Grant License Agreement"
SSH and SFTP support is provided by the JCraft JSch package, which is open source software, available under the terms of a BSD style license. The original software and related information is available at http://www.jcraft.com/jsch/.

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

OGNL is the creation of Luke Blanshard and Drew Davidson.

Copyright 1997-2003, Drew Davidson and Luke Blanshard. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Drew Davidson nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
_DAMAGE.

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Open Source Used In DNAC Platform Arica 1519

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.427 babel-plugin-transform-decorators

6.24.1

1.427.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.428 is-finite 1.1.0
1.428.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.429 debug 2.6.9 2.6.9

1.429.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.430 `indexof 0.0.1 0.0.1`

1.431 `@babel/traverse 7.16.3`

1.431.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.432 `jersey-client 2.5.1`

1.432.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free...
software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this
Open Source Used in DNAC Platform

License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. "Contributor" means each individual or entity that
creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the
Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.
3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and
make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor
or the Original Software where the Participant is the
Initial Developer) directly or indirectly infringes any
patent, then any and all rights granted directly or
indirectly to You by such Participant, the Initial
Developer (if the Initial Developer is not the Participant)
and all Contributors under Sections 2.1 and/or 2.2 of this
License shall, upon 60 days notice from Participant
terminate prospectively and automatically at the expiration
of such 60 day notice period, unless if within such 60 day
period You withdraw Your claim with respect to the
Participant Software against such Participant either
unilaterally or pursuant to a written agreement with
Participant.

6.3. In the event of termination under Sections 6.1 or 6.2
above, all end user licenses that have been validly granted
by You or any distributor hereunder prior to termination
(excluding licenses granted to You by any distributor)
shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT
(INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE
INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF
COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE
LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT
LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK
STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER
COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN
INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF
LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL
INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT
APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO
NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR
CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT
APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is
defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial
computer software" (as that term is defined at 48 C.F.R. 
252.227-7014(a)(1)) and "commercial computer software
documentation" as such terms are used in 48 C.F.R. 12.212 (Sept.
through 227.7202-4 (June 1995), all U.S. Government End Users
acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

1.433 boost 1.60

1.433.1 Available under license:

Use of this software is granted under one of the following two licenses, to be chosen freely by the user.


===============================================================================

Open Source Used In DNAC Platform Arica  1541
Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2. The MIT License

===============================================================================
Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Use, modification and distribution is subject to the Boost Software License, Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt)

The default copyright note for C++ source files reads:

// Boost.Geometry (aka GGL, Generic Geometry Library)

// Copyright (c) 2007-2011 Barend Gehrels, Amsterdam, the Netherlands.
// Copyright (c) 2008-2011 Bruno Lalande, Paris, France.
// Copyright (c) 2009-2011 Mateusz Loskot, London, UK.

// Parts of Boost.Geometry are redesigned from Geodan's Geographic Library
// (geolib/GGL), copyright (c) 1995-2010 Geodan, Amsterdam, the Netherlands.

Exceptions:
1) Major work of any author -> might change order, change date
2) Exclusive work of one author including design -> might take sole copyright

Examples:
- adapted geometry types e.g. array, Boost.Polygon
- implemented file-format e.g. WKB
- implemented specific strategy or algorithm e.g. intersections
3) Examples -> might take sole copyright
4) Tests -> might take sole copyright
5) Docs -> might take sole copyright
6) Utilities e.g. converters -> might take sole copyright

The copyright note for other (source) files as .py, Jamfiles, etc is similar
and comments are changed accordingly.
Copyright:

/+\n+\n  Copyright 1993-2002 Christopher Seiwald and Perforce Software, Inc.
\n+

This is Release 2.4 of Jam/MR, a make-like program.

License is hereby granted to use this software and distribute it freely, as long as this copyright notice is retained and modifications are clearly marked.

ALL WARRANTIES ARE HEREBY DISCLAIMED.

Some portions are also:

Copyright 2001-2006 David Abrahams.
Copyright 2002-2006 Rene Rivera.
Copyright 2003-2006 Vladimir Prus.

Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)
This is an attempt to acknowledge early contributions to the garbage collector. Later contributions should instead be mentioned in README.changes.

HISTORY -

Early versions of this collector were developed as a part of research projects supported in part by the National Science Foundation and the Defense Advance Research Projects Agency.

The garbage collector originated as part of the run-time system for the Russell programming language implementation. The first version of the
The garbage collector was written primarily by Al Demers. It was then refined and mostly rewritten, primarily by Hans-J. Boehm, at Cornell U., the University of Washington, Rice University (where it was first used for C and assembly code), Xerox PARC, SGI, and HP Labs. However, significant contributions have also been made by many others.

Some other contributors:

More recent contributors are mentioned in the modification history in README.changes. My apologies for any omissions.

The SPARC specific code was originally contributed by Mark Weiser. The Encore Multimax modifications were supplied by Kevin Kenny (kenny@m.cs.uiuc.edu). The adaptation to the IBM PC/RT is largely due to Vernon Lee, on machines made available to Rice by IBM. Much of the HP specific code and a number of good suggestions for improving the generic code are due to Walter Underwood. Robert Brazile (brazil@diamond.bbn.com) originally supplied the ULTRIX code. Al Dosser (dosser@src.dec.com) and Regis Cridlig (Regis.Cridlig@cl.cam.ac.uk) subsequently provided updates and information on variation between ULTRIX systems. Parag Patel (parag@netcom.com) supplied the A/UX code. Jesper Peterson (jep@mtiame.mtia.oz.au), Michel Schinz, and Martin Tauchmann (martintauchmann@bigfoot.com) supplied the Amiga port. Thomas Funke (thf@zelator.in-berlin.de(?)) and Brian D.Carlstrom (bdc@clark.lcs.mit.edu) supplied the NeXT ports. Douglas Steel (doug@wg.icl.co.uk) provided ICL DRS6000 code. Bill Janssen (janssen@parc.xerox.com) supplied the SunOS dynamic loader specific code. Manuel Serrano (serrano@cornas.inria.fr) supplied linux and Sony News specific code. Al Dosser provided Alpha/OSF/1 code. He and Dave Detlefs (detlefs@src.dec.com) also provided several generic bug fixes. Alistair G. Crooks (agc@uts.amdahl.com) supplied the NetBSD and 386BSD ports. Jeffrey Hsu (hsu@soda.berkeley.edu) provided the FreeBSD port. Brent Benson (brent@jade.ssd.csd.harris.com) ported the collector to a Motorola 88K processor running CX/UX (Harris NightHawk). Ari Huttunen (Ari.Huttunen@hut.fi) generalized the OS/2 port to nonIBM development environments (a nontrivial task). Patrick Beard (beard@cs.ucdavis.edu) provided the initial MacOS port. David Chase, then at Olivetti Research, suggested several improvements. Scott Schwartz (schwartz@groucho.cse.psu.edu) supplied some of the code to save and print call stacks for leak detection on a SPARC. Jesse Hull and John Ellis supplied the C++ interface code. Zhong Shao performed much of the experimentation that led to the current typed allocation facility. (His dynamic type inference code hasn’t made it into the released version of the collector, yet.)

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Software License, Version 1.0

Copyright 2002-2003, Trustees of Indiana University.
Copyright 2000-2001, University of Notre Dame.
All rights reserved.

Indiana University has the exclusive rights to license this product under the following license.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* All redistributions of source code must retain the above copyright notice, the list of authors in the original source code, this list of conditions and the disclaimer listed in this license;

* All redistributions in binary form must reproduce the above copyright notice, this list of conditions and the disclaimer listed in this license in the documentation and/or other materials provided with the distribution;

* Any documentation included with all redistributions must include the following acknowledgement:

"This product includes software developed at the University of Notre Dame and the Pervasive Technology Labs at Indiana University. For technical information contact Andrew Lumsdaine at the Pervasive Technology Labs at Indiana University. For administrative and license questions contact the Advanced Research and Technology Institute at 351 West 10th Street, Indianapolis, Indiana 46202, phone 317-278-4100, fax 317-274-5902."
Alternatively, this acknowledgement may appear in the software itself, and wherever such third-party acknowledgments normally appear.

* The name Indiana University, the University of Notre Dame or "Caramel" shall not be used to endorse or promote products derived from this software without prior written permission from Indiana University. For written permission, please contact Indiana University Advanced Research & Technology Institute.

* Products derived from this software may not be called "Caramel", nor may Indiana University, the University of Notre Dame or "Caramel" appear in their name, without prior written permission of Indiana University Advanced Research & Technology Institute.

Indiana University provides no reassurances that the source code provided does not infringe the patent or any other intellectual property rights of any other entity. Indiana University disclaims any liability to any recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise.

LICENSEE UNDERSTANDS THAT SOFTWARE IS PROVIDED "AS IS" FOR WHICH NO WARRANTIES AS TO CAPABILITIES OR ACCURACY ARE MADE. INDIANA UNIVERSITY GIVES NO WARRANTIES AND MAKES NO REPRESENTATION THAT SOFTWARE IS FREE OF INFRINGEMENT OF THIRD PARTY PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS. INDIANA UNIVERSITY MAKES NO WARRANTIES THAT SOFTWARE IS FREE FROM "BUGS", "VIRUSES", "TROJAN HORSES", "TRAP DOORS", "WORMS", OR OTHER HARMFUL CODE. LICENSEE ASSUMES THE ENTIRE RISK AS TO THE PERFORMANCE OF SOFTWARE AND/OR ASSOCIATED MATERIALS, AND TO THE PERFORMANCE AND VALIDITY OF INFORMATION GENERATED USING SOFTWARE.

The following people hereby grant permission to replace all existing licenses on their contributions to Boost with the Boost Software License, Version 1.0. (boostinspect:nolicense boostinspect:nocopyright)

Aleksey Gurtovoy (agurtovoy@meta-comm.com)
Andrei Alexandrescu (andrewalex - at - hotmail.com) (See Boost list message of August 12, 2004 11:06:58 AM EST)
Andrew Lumsdaine ()
Anthony Williams (anthony -at- justsoftwaresolutions.co.uk)
Beman Dawes (bdawes@acm.org)
Brad King (brad.king -at- kitware.com) (See Boost list message of Wed, 21 Jul 2004 11:15:46 -0400)
Brian Osman (osman -at- vvisions.com) (See CVS log)
Bruce Barr (schmoost -at- yahoo.com) (See Boost list of Mon, 16 Aug 2004 15:06:43 -0500)
Bruno da Silva de Oliveira (bruno - at - esss.com.br)
Christian Engstrom (christian.engstrom -at- glindra.org) (See Boost list message of Mon, 30 Aug 2004 14:31:49 +0200)
Cromwell D Enage (sponage -at- yahoo.com) (See Boost list message of August 12, 2004 11:49:13 AM EST)
Dan Gohman (djg -at- cray.com) (See Boost list message of Sat, 21 Aug 2004 10:54:59 +0100)
Dan Nuffer (dan -at- nuffer.name)
Daniel Frey (d.frey -at- gmx.de, daniel.frey -at- aixigo.de)
Daniel Nuffer (dan -at- nuffer.name)
Darin Adler (darin -at- bentspoon.com) (Email to Andreas Huber, see change log)
Daryle Walker (darylew - at - hotmail.com)
Dave Abrahams (dave@boost-consulting.com)
Dave Moore (dmoore -at- viefinancial.com) (See Boost list message of 18 Dec 2003 15:35:50 -0500)
David Abrahams (dave@boost-consulting.com)
Dietmar Kuehl (dietmar_kuehl -at- yahoo.com) (Email to Andreas Huber, see change log)
Douglas Gregor (gregod -at- cs.rpi.edu, dgregor -at- cs.indiana.edu, doug.gregor -at- gmail.com)
Dr John Maddock (john - at - johnmaddock.co.uk)
Edward D. Brey (brey -at- ductape.net) (Email to Andreas Huber, see change log)
Eric Ford (un506n902 -at- sneakemail.com) (See Boost list message of Sun, 15 Aug 2004 10:29:13 +0100)
Eric Friedman (ebf@users.sourceforge.net)
Eric Niebler (eric@boost-consulting.com)
Fernando Cacciola (fernando_cacciola@ciudad.com.ar)
Fernando Luis Cacciola Carballal (fernando_cacciola@ciudad.com.ar)
Francois Faure (Francois.Faure -at- imag.fr) (See CVS log)
Gary Powell (powellg - at - amazon.com) (See Boost list message of 10 Feb 2004 14:22:46 -0800)
Gennadiy Rozental (rogoeff -at- mail.com) (Email to Andreas Huber, see change log)
Gottfried Ganssaue (Gottfried.Ganssaue -at- HAUF.E.DE) (See Boost List message of Mon, 16 Aug 2004 10:09:19 +0200)
Gottfried Ganaogue (Gottfried.Gansaauge -at- HAUF.E.DE) (Alternative spelling of Gottfried Ganssauge)
Greg Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)
Gregory Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)
Gunter Winkler (gunter.winkler -at- unibw-muenchen.de) (See Boost List message of Mon, 16 Aug 2004 10:24:17 +0200)
Hartmut Kaiser (hartmut.kaiser -at- gmail.com)
Herve Bronnimann (hbr -at- poly.edu)
Herv Brnnimman (hbr -at- poly.edu)
Housemarque Oy (Ilari Kuittinen ilari.kuittinen -at- housemarque.fi)
Howard Hinnant (hinnant -at- twcny.rr.com) (See Boost list message of July 25, 2004 3:44:49 PM EST)
Hubert Holin (hubert_holin -at- users.sourceforge.net)
Indiana University ()
Itay Maman (imaman -at- users.sourceforge.net)
Jaakko Jrvi (jajarvi -at- osl.iu.edu)
Jaap Suter (j.suter -at- student.utwente.nl) (See Boost list message of Thu, 16 Sep 2004 09:32:43 -0700)
Jeff Garland (jeff - at - crystalclearsoftware.com) (see Boost list post of July 25, 2004 19:31:09 -0700)
Jens Maurer (Jens.Maurer@gmx.net)
Jeremy G Siek (jsiek@osl.iu.edu)
Jeremy Siek (jsiek@osl.iu.edu)
Joel de Guzman (joel -at- boost-consulting.com) (See Boost list message of July 25, 2004 8:32:00 PM EST)
John Bandela (jbandela-at-ufl.edu)
John Maddock (john - at - johnmaddock.co.uk)
John R Bandela (jbandela-at-ufl.edu)
Jonathan Turkanis (turkanis -at- corderage dot com)
Juergen Hunold (hunold -at- ive.uni-hannover.de) (See Boost List Message of Fri, 13 Aug 2004 19:39:55 +0200)
Kevlin Henney (kevlin -at- curbralan.com) (See Boost list message of Wed, 15 Sep 2004 18:15:17 +0200)
Kresimir Fresl (fresl -at- master.grad.hr) (See Boost List message of August 16, 2004 8:23:35 AM EST)
Lars Gullik Bjønes (larsbj -at- lyx.org) (See Boost list message of Tue, 17 Aug 2004 15:49:02 +0100)
Lie-Quan Lee (liequan - at - slac.stanford.edu, llee - at - cs.indiana.edu)
Maarten Keijzer (mkeijzer -at- cs.vu.nl) (See Boost list message of Wed, 18 Aug 2004 21:43:18 +0100)
Mac Murrett (mmurrett -at- mac.com)
Marc Wintermantel (wintermantel -at- imes.mavt.ethz.ch, wintermantel -at- even-ag.ch) (See CVS log)
Michael Glassford (glassfordm - at - hotmail.com)
Michael Stevens (Michael.Stevens - at - epost.de)
Multi Media Ltd. (pdimov@mmltd.net)
Nicolai M Josuttis (solutions -at- josuttis.com) (See Boost list message of Mon, 30 Aug 2004 10:52:00 +0100)
Nikolay Mladenov (nickm - at - sitius.com) (See Boost list message of Tue, 17 Aug 2004 15:45:33 +0100)
Pavol Droba (droba -at- topmail.sk)
Peter Dimov (pdimov@mmltd.net)
R W Grosse-Kunstleve (RWGrosse-Kunstleve@lbl.gov)
Ralf W. Grosse-Kunstleve (RWGrosse-Kunstleve@lbl.gov)
Rational Discovery LLC (Greg Landrum Landrum -at- RationalDiscovery.com) (See Boost list post of Tue, 17 Aug 2004 10:35:36 +0100)
Rene Rivera (grafik/redshift-software.com, rrivera/acm.org)
Robert Ramey (ramey@www.rrsd.com)
Roland Richter (roland -at- flll,jku.at) (See Boost list post of Mon, 16 Aug 2004 22:16:55 +0200)
Roland Schwarz (roland.schwarz -at- chello.at)
Ronald Garcia (garcia -at- cs.indiana.edu) (Email to Andreas Huber, see change log)
Samuel Krempf (krempf -at- crans.ens-cachan.fr) (See Boost list message of Mon, 27 Sep 2004 13:18:36 +0200)
Stefan Seefeld (seefeld -at- sympatico.ca)
Stephen Cleary (sceary -at- jerviswebb.com) (See Boost list message of Tue, 28 Sep 2004 13:11:46 +0100)
Steve Cleary (Variant of Stephen Cleary)
Sylvain Pion (Sylvain.Pion - at - sophia.inria.fr)
The Trustees of Indiana University ()
The Trustees of Indiana University ()
Thomas Witt (witt - at - ive.uni-hannover.de, witt - at - acm.org, witt - at - styleadvisor.com)
Thorsten Jrgen Ottosen (nesotto - at - cs.auc.dk)
Thorsten Ottosen (nesotto - at - cs.auc.dk)
Toon Knapen (toon dot knapen - at - fft.be)
The Trustees of Indiana University ()
University of Notre Dame ()
Vladimir Prus (ghost@cs.msu.su)
William E. Kempf () (email to Beman Dawes, 9/14/2006 4:18 PM)
Joerg Walter (jhr.walter - at - t-online.de : email to ublas mailing list Mon, 17 Sep 2007 10:17:08 +0200)
Mathias Koch (mkoch - at - idesis.de 7 : email to boost-owner@lists.boost.org Sep 2007 13:20:09 +0200)

--- end ---

1.434 boxen 5.1.2
1.434.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.


1.435.1 Available under license: The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.436 bson 3.12.4

1.436.1 Available under license: No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008-present MongoDB, Inc.

* License under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/LocalDateCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/InstantCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/LocalTimeCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/Jsr310CodecProvider.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/LocalDateTimeCodec.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2008-present MongoDB, Inc.

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/BsonCodecProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2010 The Guava Authors
 * Copyright 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/internal/UnsignedLongs.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):

* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/OverridableUuidRepresentationCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/JsonParseException.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/RelaxedExtendedJsonDoubleConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/annotations/BsonDiscriminator.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/BsonMinKey.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/ClassModelBuilder.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/ObjectId.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/ExtendedJsonBinaryConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/NoOpFieldNameValidator.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/jsr310/DateTimeBasedCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/property/PropertyMetadata.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/HeaderCodecRegistry.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/PropertyModelSerializationImpl.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/internal/ProvidersCodecRegistry.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/ShellUndefinedConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/internal/CycleDetectingCodecRegistry.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/OverridableUuidRepresentationCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/UuidCodecHelper.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/LegacyExtendedJsonDateTimeConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/RawBsonDocument.java
Open Source Used In DNAC Platform Arica 1554
jar/org/bson/codecs/pojo/PojoCodecProvider.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/ExtendedJsonMaxKeyConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/BsonDateTimeCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/FallbackPropertyCodecProvider.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/PropertyCodecProvider.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/CodeCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/DecoderContext.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/AtomicIntegerCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/BinaryCodec.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/PojoCodecRegistry.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/ShellTimestampConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/ShellBinaryConverter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/pojo/PropertyCodecRegistrar.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/package-info.java
jar/org/bson/codecs/configuration/CodecProvider.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/BSOAN.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/internal/UuidHelper.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/json/StrictCharacterStreamJsonWriter.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/BsonBinarySubType.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/codecs/UuidCodec.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright (c) 2008-2014 Atlassian Pty Ltd
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/util/CopyOnWriteMap.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/util/AbstractCopyOnWriteMap.java
* /opt/cola/permits/1136447680_1613767071.02/0/bson-3-12-4-sources-2-jar/org/bson/util/AbstractCopyOnWriteMap.java

1.437 moment 2.27.0 2.27.0

1.437.1 Available under license :
Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.438 ansi-styles 1.0.0 1.0.0

1.438.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.439 memory 0.9.0

1.439.1 Available under license :

Arrow Memory
Copyright 2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.440 http-errors 1.3.1 1.3.1
1.440.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.441 twill-common 0.6.0-incubating
1.441.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/filesystem/LocationFactories.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/common/Threads.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/filesystem/ForwardingLocationFactory.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/filesystem/LocalLocation.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/common/Cancellable.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/filesystem/LocalLocationFactory.java
* /opt/cola/permits/1137746870_1614121577.24/0/twill-common-0-6-0-incubating-sources-2-jar/org/apache/twill/filesystem/Location.java

1.442 path-exists 3.0.0
1.442.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.443 @babel/plugin-proposal-object-rest-spread 7.16.0

1.443.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.444 camelize 1.0.0 1.0.0

1.444.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.445 @babel/helper-annotate-as-pure 7.16.0
1.445.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.446 engine.io-parser 1.3.2 1.3.2
1.446.1 Available under license:
(The MIT License)

Copyright (c) 2016 Guillermo Rauch (@rauchg)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ‘AS IS’, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.447 @babel/plugin-transform-literals 7.16.0

1.447.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.448 curator-client 5.1.0

Open Source Used In DNAC Platform Arica 1572
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.449 babel-helper-create-regexp-features-plugin 7.16.0

1.449.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.450 hbase-shaded-miscellaneous 3.4.1
1.450.1 Available under license:

Apache Commons Collections
Copyright 2001-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache HBase Relocated (Shaded) Third-party Miscellaneous Libs
Copyright 2017-2020 The Apache Software Foundation

Apache Commons CLI
Copyright 2001-2017 The Apache Software Foundation

1.451 babel-plugin-transform-es2015-block-scoped-functions 6.22.0
1.451.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGligence OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.452 fast-infoset 1.2.15

1.453 jkeylockmanager 2.1.0

1.453.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
you Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the

Open Source Used In DNAC Platform Arica 1584
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.454 babel-plugin-transform-async-to-generator 6.24.1 6.24.1
Available under license:

**1.454.1 Available under license:**

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.455 amqplib 0.5.2 0.5.2**

**1.455.1 Available under license:**

amqplib copyright (c) 2013, 2014

Michael Bridgen <mikeb@squaremobius.net>

This package, "amqplib", is licensed under the MIT License. A copy may be found in the file LICENSE-MIT in this directory, or downloaded from http://opensource.org/licenses/MIT.

The MIT License (MIT)

Copyright (c) 2013, 2014 Michael Bridgen <mikeb@squaremobius.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.456 get-value 2.0.6 2.0.6

1.456.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.457 lodash.reduce 4.6.0 4.6.0

1.457.1 Available under license:
Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.458 open-csv 2.3

1.458.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2007 Kyle Miller.
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
* <p/>
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/bean/HeaderColumnNameMappingStrategy.java
No license file was found, but licenses were detected in source scan.

A very simple CSV parser for Java released under a commercial-friendly license.

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/package.html
No license file was found, but licenses were detected in source scan.

/**
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/ResultSetHelperService.java
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/ResultSetHelper.java
No license file was found, but licenses were detected in source scan.

/**
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/**
/** *
* A very simple CSV writer released under a commercial-friendly license.
* *
* @author Glen Smith
* *
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
jar/au/com/bytecode/opencsv/CSVWriter.java
No license file was found, but licenses were detected in source scan.

/**
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/**
/** *
* A very simple CSV parser released under a commercial-friendly license.
* This just implements splitting a single line into fields.
* *
* @author Glen Smith
* @author Rainer Pruy
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
jar/au/com/bytecode/opencsv/CSVParser.java
No license file was found, but licenses were detected in source scan.

/**
Copyright 2007 Kyle Miller.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/bean/MappingStrategy.java
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/bean/ColumnPositionMappingStrategy.java
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2-3-sources-3-
  jar/au/com/bytecode/opencsv/bean/CsvToBean.java

No license file was found, but licenses were detected in source scan.

/**
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

/**
* A very simple CSV reader released under a commercial-friendly license.
*
* @author Glen Smith
*
*/
Open Source Used In DNAC Platform Arica  1592

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2.3-sources-3-
  jar/au/com/bytecode/opencsv/CSVReader.java
No license file was found, but licenses were detected in source scan.

/**
Copyright 2007,2010 Kyle Miller.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206716_1613708914.54/0/opencsv-2.3-sources-3-
  jar/au/com/bytecode/opencsv/bean/HeaderColumnNameTranslateMappingStrategy.java

1.459 de.huxhorn.lilith.logback.classic 0.9.44
1.459.1 Available under license :
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall
apply, that proxy’s public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.
/*
* Lilith - a log event viewer.
* Copyright (C) 2007-2011 Joern Huxhorn
*
* This program is free software: you can redistribute it and/or modify
* it under the terms of the GNU Lesser General Public License as published by
* the Free Software Foundation, either version 3 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU Lesser General Public License for more details.
*
* You should have received a copy of the GNU Lesser General Public License
* along with this program. If not, see <http://www.gnu.org/licenses/>.
* /
*
* Copyright 2007-2011 Joern Huxhorn
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /

1.460 xtend 4.0.2 4.0.2
1.460.1 Available under license:
The MIT License (MIT)
Copyright (c) 2012-2014 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.461 next-tick 1.0.0

1.461.1 Available under license:

The MIT License

Copyright (C) 2012-2016 Mariusz Nowak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.462 glew 1.3.1-1ubuntu0.16.04.2

1.462.1 Available under license:

The OpenGL Extension Wrangler Library

Copyright (C) 2002-2007. Milan Ikits <milan ikits[ ]ieee org>
Copyright (C) 2002-2007. Marcelo E. Magallon <mmagallo[ ]debian org>
Copyright (C) 2002. Lev Povalahev
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The name of the author may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Mesa 3-D graphics library
Version: 7.0

Copyright (C) 1999-2007 Brian Paul All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

/*
 ** Copyright (c) 2007 The Khronos Group Inc.
 **
 ** Permission is hereby granted, free of charge, to any person obtaining a
 ** copy of this software and/or associated documentation files (the
 ** "Materials"), to deal in the Materials without restriction, including
 ** without limitation the rights to use, copy, modify, merge, publish,
 ** distribute, sublicense, and/or sell copies of the Materials, and to
 ** permit persons to whom the Materials are furnished to do so, subject to
 ** the following conditions:
 **
 ** The above copyright notice and this permission notice shall be included
 ** in all copies or substantial portions of the Materials.
 **
 ** THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 ** EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 ** MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 ** IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 ** CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 ** TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 ** MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
 */
/*
 * Mesa 3-D graphics library
 * Version: 7.0
 *
 * Copyright (C) 1999-2007 Brian Paul All Rights Reserved.
* Permission is hereby granted, free of charge, to any person obtaining a
* copy of this software and associated documentation files (the "Software"),
* to deal in the Software without restriction, including without limitation
* the rights to use, copy, modify, merge, publish, distribute, sublicense,
* and/or sell copies of the Software, and to permit persons to whom the
* Software is furnished to do so, subject to the following conditions:
* 
* The above copyright notice and this permission notice shall be included
* in all copies or substantial portions of the Software.
* 
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
* OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
* BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year  name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

/*
** The OpenGL Extension Wrangler Library
** Copyright (C) 2008-2017, Nigel Stewart <nigels[users sourceforge net>
** Copyright (C) 2002-2008, Milan Ikits <milan ikits[IEEE org>
** Copyright (C) 2002-2008, Marcelo E. Magallon <mmagallo[Debian org>
** Copyright (C) 2002, Lev Povalahev
** All rights reserved.
**
** Redistribution and use in source and binary forms, with or without
** modification, are permitted provided that the following conditions are met:
**
** * Redistributions of source code must retain the above copyright notice,
**   this list of conditions and the following disclaimer.
** * Redistributions in binary form must reproduce the above copyright notice,
**   this list of conditions and the following disclaimer in the documentation
**   and/or other materials provided with the distribution.
** * The name of the author may be used to endorse or promote products
**   derived from this software without specific prior written permission.
**
** THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
** AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
** IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
** ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
** LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
** CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
** SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
** INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN ** CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ** ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF ** THE POSSIBILITY OF SUCH DAMAGE.
*/

1.463 cliui 5.0.0 5.0.0

1.463.1 Available under license:

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.464 tephra-core 0.6.0

1.464.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addition to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by
brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and
description of purpose be included on the same "printed page" as the copyright notice for easier identification within
third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.465 drill-java-exec 1.19.0
1.465.1 Available under license:

Drill : Exec : Java Execution Engine
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.466 jws 3.2.2 3.2.2

1.466.1 Available under license:

Copyright (c) 2013 Brian J. Brennan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.467 finalhandler 1.1.1 1.1.1

1.467.1 Available under license:
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.468 ignore 3.3.10

1.468.1 Available under license:
MIT

1.469 is-dotfile 1.0.3

1.469.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.470 chardet 0.7.0 0.7.0
1.470.1 Available under license:
Copyright (C) 2018 Dmitry Shirokov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.471 swagger-jaxrs 1.5.0

1.472 extsprintf 1.4.0 1.4.0
1.472.1 Available under license:
Copyright (c) 2012, Joyent, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE

1.473 react-copy-to-clipboard 5.0.2 5.0.2

1.473.1 Available under license:

MIT License

Copyright (c) 2017 Mauro Bringolf

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2016 John Hiesey

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2008, Fair Oaks Labs, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation.
and/or other materials provided with the distribution.

* Neither the name of Fair Oaks Labs, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright JS Foundation and other contributors, https://js.foundation
Copyright (C) 2012-2013 Yusuke Suzuki (twitter: @Constellation) and other contributors.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
 * Copyright (c) 2010, Linden Research, Inc.
 * Copyright (c) 2012, Joshua Bell
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2017 CoderPuppy
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-2018 Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ISC License

Copyright (c) npm, Inc.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE COPYRIGHT HOLDER DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright 2018 Kilian Valkhof

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
(The MIT License)

Copyright (c) 2012-2014 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supereede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License (MIT)

Copyright (c) 2012 Raynos.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/*! ****************************************************************************
Copyright (c) Microsoft Corporation. All rights reserved.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

THIS CODE IS PROVIDED ON AN *AS IS* BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED

Open Source Used In DNAC Platform Arica 1630
																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
WARRANTIES OR CONDITIONS OF TITLE, FITNESS FOR A PARTICULAR PURPOSE,
MERCHANTABILITY OR NON-INFRINGEMENT.

See the Apache Version 2.0 License for specific language governing permissions
and limitations under the License.
***************************************************************************** */

The MIT License (MIT)

Copyright (c) 2011 Troy Goode <troygoode@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014 Tim Schaub

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
(The MIT License)

Copyright (c) 2012-2013 TJ Holowaychuk
Copyright (c) 2015 Andreas Lubbe
Copyright (c) 2015 Tiancheng "Timothy" Gu

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2012 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

You may use the Forge project under the terms of either the BSD License or the GNU General Public License (GPL) Version 2.

The BSD License is recommended for most projects. It is simple and easy to
understand and it places almost no restrictions on what you can do with the Forge project.

If the GPL suits your project better you are also free to use Forge under that license.

You don't have to do anything special to choose one license or the other and you don't have to notify anyone which license you are using. You are free to use this project in commercial projects as long as the copyright header is left intact.

If you are a commercial entity and use this set of libraries in your commercial software then reasonable payment to Digital Bazaar, if you can afford it, is not required but is expected and would be appreciated. If this library saves you time, then it's saving you money. The cost of developing the Forge software was on the order of several hundred hours and tens of thousands of dollars. We are attempting to strike a balance between helping the development community while not being taken advantage of by lucrative commercial entities for our efforts.

New BSD License (3-clause)
Copyright (c) 2010, Digital Bazaar, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Digital Bazaar, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DIGITAL BAZAAR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

ISC License

Copyright (c) 2018, Andrea Giammarchi, @WebReflection

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

=======================================

Copyright (c) 2014 LevelUP contributors

*LevelUP contributors listed at <https://github.com/rvagg/node-levelup#contributors>*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2019 Lus Couto <hello@luiscouto.pt>

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2017 Sergey Rubanov (https://github.com/chicoxyzzy)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017, 2018 Simon Lydell
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Nathan Rajlich <nathan@tootallnate.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

MIT License
Copyright (c) Kevin Mrtensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2013-2017 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 crypto-browserify

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This software is released under the MIT license:

Copyright (c) James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-present, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The MIT License (MIT)

Copyright (c) 2016 Nik Butenko

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2013 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Kevin Martensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2015-2018 Google, Inc., Netflix, Inc., Microsoft Corp. and contributors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
The MIT License (MIT)

Copyright (c) 2015 Roy Riojas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2011 Einar Otto Stangvik <einaros@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright 2014 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 sudodoki <smd.deluzion@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 webpack-contrib

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Toposort - Topological sorting for node.js
Copyright (c) 2012 by Marcel Klehr <mklehr@gmx.net>
MIT LICENSE

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)

Copyright 2013 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(MIT License)
Copyright (c) 2013 [Ramesh Nair](http://www.hiddentao.com/)

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 Gustaf Andersson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2015 Tim Caswell (https://github.com/creationix) and other
contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Some files from the tests folder are from joyent/node and mscedex/io.js, a fork
of nodejs/io.js:

- tests/iojs/test-http-parser-durability.js

This file is from https://github.com/mscedex/io.js/blob/js-http-parser/test/pummel/test-http-parser-durability.js
with modifications by Jan Schr (jscisssr).

---

Copyright io.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

---

- tests/fixtures/*
These files are from https://github.com/nodejs/node with changes by Jan Schr (jscissr).

Node.js is licensed for use as follows:

---

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

---

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014 Forbes Lindesay

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2014 Marten de Vries

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2012 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com
Copyright (c) 2016 Douglas Christopher Wilson doug@somethingdoug.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2017 Kat Marchn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2016 Airbnb
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013-2018 Ben Alman <cowboy@rj3.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

UglifyJS is released under the BSD license:

Copyright 2012-2018 (c) Mihai Bazon <mihai.bazon@gmail.com>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This package contains code originally written by Isaac Z. Schlueter.
Used with permission.

Copyright (c) Meryn Stol ("Author")
All rights reserved.

The BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
(The MIT License)

Copyright (c) 2012 - 2015 Tobias Koppers

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE.

MIT License

Copyright (c) 2013 JosF. Romaniello

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE.

The MIT License (MIT)
Copyright 2015 Kiko Beats

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Wes Todd

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) Bogdan Chadkin <trysound@yandex.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Javier Blanco

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Espree
Copyright JS Foundation and other contributors, https://js.foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License
Copyright (c) 2017 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Andres Suarez <zertosh@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) npm, Inc. and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Kevin Gravier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License

Copyright (c) Tobias Koppers @sokra

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the 'Software'), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(ISC License (ISC)
Copyright 2018 Glen Maddern

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby
grant, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2017 JS Foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014, Rebecca Turner <me@re-becca.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License

Copyright (c) 2013 James Halliday (mail@substack.net)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice
shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (C) 2012 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 JD Ballard

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015, 2017-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2014-2017 Calvin Metcalf, Fedor Indutny & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# spdx-license-ids


A list of [SPDX license](https://spdx.org/licenses/) identifiers
## Installation

[Download JSON directly](https://raw.githubusercontent.com/shinnn/spdx-license-ids/master/index.json), or [use](https://docs.npmjs.com/cli/install) [npm](https://docs.npmjs.com/about-npm):

```javascript
npm install spdx-license-ids
```

## [Node.js](https://nodejs.org/) API

### require('spdx-license-ids')

Type: `string[]`

All license IDs except for the currently deprecated ones.

```javascript
const ids = require('spdx-license-ids');
//=> ['0BSD', 'AAL', 'ADSL', 'AFL-1.1', 'AFL-1.2', 'AFL-2.0', 'AFL-2.1', 'AFL-3.0', 'AGPL-1.0-only', ...]

ids.includes('BSD-3-Clause'); //=> true
ids.includes('CC-BY-1.0'); //=> true

ids.includes('GPL-3.0'); //=> false
```

### require('spdx-license-ids/deprecated')

Type: `string[]`

Deprecated license IDs.

```javascript
const deprecatedIds = require('spdx-license-ids/deprecated');
//=> ['AGPL-1.0', 'AGPL-3.0', 'GFDL-1.1', 'GFDL-1.2', 'GFDL-1.3', 'GPL-1.0', 'GPL-2.0', ...]

deprecatedIds.includes('BSD-3-Clause'); //=> false
deprecatedIds.includes('CC-BY-1.0'); //=> false

deprecatedIds.includes('GPL-3.0'); //=> true
```

## License

[Creative Commons Zero v1.0 Universal](https://creativecommons.org/publicdomain/zero/1.0/deed)
Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE
FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors
Copyright 2015 Calvin Metcalf

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.
MIT License

Copyright (c) 2012, 2013, 2014 James Halliday <mail@substack.net>, 2009 Thomas Robinson <280north.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013 Ryan Day

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-2018 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright JS Foundation and other contributors, https://js.foundation/

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2016 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Daniel Cousens

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
Copyright 2014 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the Yahoo! Inc. nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Apache License, Version 2.0

Copyright (c) 2011 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright (c) 2014-2016 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) Fork, Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2012 Simon Boudrias

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

This project contains parts of Node.js.
Node.js is licensed for use as follows:

====
Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

====

This license applies to parts of Node.js originating from the
https://github.com/joyent/node repository:
Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

License

(The MIT License)

Copyright (c) 2014 The cheeriojs contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2016 Ethan Cohen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2011 Alexander Shtuchkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2012 Ben Ripkens http://bripkens.de

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
Copyright (c) 2017 crypto-browserify contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Jon Harris

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2015 Calvin Metcalf and browserify-sign contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above
copyt

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) Ben Drucker <bvdrucker@gmail.com> (bendrucker.me)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Jed Watson <jed.watson@me.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation.
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT Licence (Expat).

Copyright (c) 2018 Michael Mclaughlin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2013 Max Ogden

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013-2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

MIT License

Copyright (c) Sam Verschueren <sam.verschueren@gmail.com> (github.com/SamVerschueren)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions.
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2016 Matthew Davidson

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2015 Kyle E. Mitchell & other authors listed in AUTHORS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Sam Verschueren <sam.verschueren@gmail.com> (github.com/SamVerschueren)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2012-2018 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
node-http-proxy

Copyright (c) 2010-2016 Charlie Robbins, Jarrett Cruger & the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2010-2018 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

node-portfinder

Copyright (c) 2012 Charlie Robbins

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Copyright (c) 2018 Toru Nagashima

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
MIT License

Copyright (c) 2017

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2013, Dominic Tarr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.
(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013 Mikola Lysenko

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2019 Andres Suarez

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2016-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright 2014-2019 Benjamin Tan <https://bnjmnt4n.now.sh/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
“Software”), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Nikita Gusakov

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
At your option you may choose either of the following licenses:

* The MIT License (MIT)
* The Apache License 2.0 (Apache-2.0)
The MIT License (MIT)

Copyright (c) 2015 AJ ONeal

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

Open Source Used In DNAC Platform Arica 1708
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 AJ ONeal

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License (MIT)
Copyright (c) 2017 Samuel Reed <samuel.trace.reed@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (C) 2018 Dmitry Shirokov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2010-2019 James Coglan

Licensed under the Apache License, Version 2.0 (the "License"); you may not use
this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License (MIT)

Copyright (c) 2013 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This software is released under the MIT license:

Copyright (c) 2012-2015 Thorsten Lorenz

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The ISC License

Copyright npm, Inc

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
(The MIT License)

Copyright (c) 2012-2014 Roman Shtylman <shtylman@gmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Copyright (c) 2018 Sven Sauleau <sven@sauleau.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copy of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Tim Perry

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) Steven Vachon <contact@svachon.com> (svachon.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (C) 2011-2015 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)
Copyright (c) 2012-2014 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Rebecca Turner <me@re-becca.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

(The MIT License)

Copyright (c) 2014 Component contributors <dev@component.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

# timers-browserify

This project uses the [MIT](http://jryans.mit-license.org/) license:

Copyright 2012 J. Ryan Stinnett <jryans@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# lib/node

The `lib/node` directory borrows files from joyent/node which uses the following license:

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013-2018, Viacheslav Lotsmanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2016 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Parts of the code are derived from various open source projects.

For code derived from Socket.IO by Guillermo Rauch see https://github.com/LearnBoost/socket.io/tree/0.6.17#readme.

All other code is released on MIT license, see LICENSE.
The MIT License (MIT)

Copyright (c) 2016 Thomas Watson Steen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
Minizlib was created by Isaac Z. Schlueter.
It is a derivative work of the Node.js project.

---

Copyright Isaac Z. Schlueter and Contributors
Copyright Node.js contributors. All rights reserved.
Copyright Joyent, Inc. and other Node contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

(The MIT License)

Copyright (c) 2011 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
Copyright (c) 2012 Tobias Koppers

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Ilya Radchenko <ilya@burstcreations.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2018 NAN contributors

*NAN contributors listed at <https://github.com/nodejs/nan#contributors>*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN
ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES
REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR
DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE
COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY
COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS
AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and
to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor’s copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., “French translation of the Work by Original Author,” or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any
Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.
8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this
license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at http://creativecommons.org/.

(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Fedor Indutny, 2015.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## Glob Logo

Glob's logo created by Tanya Brassie <http://tanyabrassie.com/>, licensed
under a Creative Commons Attribution-ShareAlike 4.0 International License
https://creativecommons.org/licenses/by-sa/4.0/

The MIT License

Copyright (c) EventSource GitHub organisation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Kevin Mrtensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Kirill Fomichev
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
(The MIT License)

Copyright (c) 2013-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2014-2019 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add your own attribution notices within Derivative Works that you distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add your own copyright statement to your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of your modifications, or for any such Derivative Works as a whole, provided your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless you explicitly state otherwise, any contribution intentionally submitted for inclusion in the Work by you to the licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with licensor regarding such contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, licensor provides the Work (and each contributor provides its contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any contributor be liable to you for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) 2015, 2019 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

MIT License

Copyright (c) 2014-2018 Suguru Motegi
Based on Async.js, Copyright Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2015 Glen Maddern

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (C) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE SOFTWARE OR THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2012 Barnesandnoble.com, llc, Donavon West, and Domenic Denicola

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT Expat Licence.

Copyright (c) 2012 Michael Mclaughlin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2016, 2018 Linus Unnebeck

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Ben Mosher

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Thiago de Arruda

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright npm, Inc

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2018 Made With MOXY Lda <hello@moxy.studio>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2014-present Olivier Lalonde <olalonde@gmail.com>, James Talmage <james@talmage.io>, Ruben
Verborgh

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR
IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

Copyright (c) 2011:
Tim Koschitzki (tim@debuggable.com)
Felix Geisendrfer (felix@debuggable.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2017 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (C) 2013 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright 2015 Mark Dalgleish <mark.john.dalgleish@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) Julian Viereck and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
Copyright (c) 2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2011-2017 JP Richardson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Arnout Kazemier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2016-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This software is dual-licensed under the ISC and MIT licenses.  
You may use this software under EITHER of the following licenses.

---------

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---------

Copyright Isaac Z. Schlueter and Contributors
All rights reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2013 Aria Minaei

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (C) 2014-2017 by Vitaly Puzrin and Andrei Tuputcyn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Attribution 4.0 International

================================================================================

Creative Commons Corporation ("Creative Commons") is not a law firm and does not provide legal services or legal advice. Distribution of Creative Commons public licenses does not create a lawyer-client or other relationship. Creative Commons makes its licenses and related information available on an "as-is" basis. Creative Commons gives no warranties regarding its licenses, any material licensed under their terms and conditions, or any related information. Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible.

Using Creative Commons Public Licenses

Creative Commons public licenses provide a standard set of terms and conditions that creators and other rights holders may use to share original works of authorship and other material subject to copyright and certain other rights specified in the public license below. The following considerations are for informational purposes only, are not exhaustive, and do not form part of our licenses.

Considerations for licensors: Our public licenses are intended for use by those authorized to give the public permission to use material in ways otherwise restricted by copyright and certain other rights. Our licenses are irrevocable. Licensors should read and understand the terms and conditions of the license they choose before applying it. Licensors should also secure all rights necessary before applying our licenses so that the public can reuse the material as expected. Licensors should clearly mark any
material not subject to the license. This includes other CC-licensed material, or material used under an exception or limitation to copyright. More considerations for licensors: wiki.creativecommons.org/Considerations_for_licensors

Considerations for the public: By using one of our public licenses, a licensor grants the public permission to use the licensed material under specified terms and conditions. If the licensor's permission is not necessary for any reason—for example, because of any applicable exception or limitation to copyright—then that use is not regulated by the license. Our licenses grant only permissions under copyright and certain other rights that a licensor has authority to grant. Use of the licensed material may still be restricted for other reasons, including because others have copyright or other rights in the material. A licensor may make special requests, such as asking that all changes be marked or described. Although not required by our licenses, you are encouraged to respect those requests where reasonable. More considerations for the public:
wiki.creativecommons.org/Considerations_for_licensees

=======================================================================

Creative Commons Attribution 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 -- Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.
b. Adapter's License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

f. Licensed Material means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

g. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. Licensor means the individual(s) or entity(ies) granting rights under this Public License.

i. Share means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.
k. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

Section 2 -- Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

   a. reproduce and Share the Licensed Material, in whole or in part; and

   b. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, simply making modifications authorized by this Section 2(a)(4) never produces Adapted Material.

5. Downstream recipients.

   a. Offer from the Licensor -- Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

   b. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or
apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.

3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 -- License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

1. If You Share the Licensed Material (including in modified form), You must:

   a. retain the following if it is supplied by the Licensor with the Licensed Material:

      i. identification of the creator(s) of the Licensed Material and any others designated to receive
attribute, in any reasonable manner requested by
the Licensor (including by pseudonym if
designated);

ii. a copyright notice;

iii. a notice that refers to this Public License;

iv. a notice that refers to the disclaimer of
warranties;

v. a URI or hyperlink to the Licensed Material to the
extent reasonably practicable;

b. indicate if You modified the Licensed Material and
retain an indication of any previous modifications; and

c. indicate the Licensed Material is licensed under this
Public License, and include the text of, or the URI or
hyperlink to, this Public License.

2. You may satisfy the conditions in Section 3(a)(1) in any
reasonable manner based on the medium, means, and context in
which You Share the Licensed Material. For example, it may be
reasonable to satisfy the conditions by providing a URI or
hyperlink to a resource that includes the required
information.

3. If requested by the Licensor, You must remove any of the
information required by Section 3(a)(1)(A) to the extent
reasonably practicable.

4. If You Share Adapted Material You produce, the Adapter's
License You apply must not prevent recipients of the Adapted
Material from complying with this Public License.

Section 4 -- Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that
apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right
to extract, reuse, reproduce, and Share all or a substantial
portion of the contents of the database;

b. if You include all or a substantial portion of the database
contents in a database in which You have Sui Generis Database
Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and

c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 -- Disclaimer of Warranties and Limitation of Liability.

a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE. WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.

b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.

c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 -- Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under
Section 6(a), it reinstates:

1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

2. upon express reinstatement by the Licensor.

For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 -- Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.

b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 -- Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.
d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

=======================================================================

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the Licensor. The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

Creative Commons may be contacted at creativecommons.org.

The MIT License (MIT)

Copyright (c) 2015 Daniel Cousens

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2011-2018 The sockjs-client Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright 2015 Djordje Lukic, Junle Li

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
/MIT License

Copyright (c) 2012-2018 Aseem Kishore, and [others].

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[others]: https://github.com/json5/json5/contributors

The MIT License (MIT)

Copyright (c) 2014 Evan Wallace

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014 Petka Antonov
2015 Sindre Sorhus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2011-2017 whitequark <whitequark@whitequark.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

MIT License
-----------

Copyright (C) 2010-2014 Philipp Dunkel

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The ISC License

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software
for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice
appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES

Open Source Used In DNAC Platform Arica 1776
OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2015, Rebecca Turner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

# The MIT License

Copyright (c) 2014-2017 James Coglan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Fog Creek Software

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<!-- LICENSEFILE/ -->

<h1>License</h1>

Unless stated otherwise all works are:

<ul><li>Copyright &copy; 2013+ <a href="http://bevry.me">Bevry Pty Ltd</a></li></ul>

and licensed under:

<ul><li><a href="http://spdx.org/licenses/MIT.html">MIT License</a></li></ul>
**MIT License**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
extcept as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

MIT License

Copyright (c) 2018 Nikita Skovoroda <chalkerx@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
Copyright (c) 2012-2015 Kit Cambridge.
http://kitcambridge.be/

Copyright (c) 2013-2015 Benjamin Tan.
https://d10.github.io/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright contributors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.
Copyright (c) 2015, Rebecca Turner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby
granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.
ISC License (ISC)
Copyright 2018 Stefan Penner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby
granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.
MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2012-2017 by Ingvar Stepanyan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## The MIT License (MIT) ##

Copyright (c) 2014 Hugh Kennedy

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2013 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2012-2016 Aseem Kishore, and [others](https://github.com/aseemk/json5/contributors).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Kael Zhang <i@kael.me>, contributors
http://kael.me/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2010 James Halliday (mail@substack.net)
Modified work Copyright 2014 Contributors (ben@npmjs.com)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2018, Gajus Kuizinas (http://gajus.com/)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Gajus Kuizinas (http://gajus.com/) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ANUARY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2013 Julian Gruber <julian@juliangruber.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016, 2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Thomas Watson Steen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Aria Minaei

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# The MIT License (MIT)

**Copyright (c) Rod Vagg (the "Original Author") and additional contributors**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2018 Tyler Kellen <tyler@sleekcode.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) DC <threedeecee@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2012 Irakli Gozalishvili. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample
code is defined as all source code displayed within the prose of the
documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own
licenses; we recommend you read them, as their terms may differ from the
terms above.
The MIT License (MIT)

Copyright (c) 2013-2018 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2012 Ryan Day

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The ISC License
Copyright (c) 2014 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2010-2018 Juriy "kangax" Zaytsev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Dual licensed under WTFPL and MIT:

---

Copyright 20132016 Domenic Denicola <d@demonic.me>

This work is free. You can redistribute it and/or modify it under the terms of the Do What The Fuck You Want To Public License, Version 2, as published by Sam Hocevar. See below for more details.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004
Copyright (C) 2004 Sam Hocevar <sam@hocevar.net>

Everyone is permitted to copy and distribute verbatim or modified copies of this license document, and changing it is allowed as long as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

---

The MIT License (MIT)

Copyright 20132016 Domenic Denicola <d@domic.me>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Doctrine
Copyright jQuery Foundation and other contributors, https://jquery.org/

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE

The MIT License (MIT)

Copyright (c) 2014-2017 createECDH contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2017, Jon Schlinkert
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2010-2016 Robert Kieffer and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
(The MIT License)

Copyright (c) 2009-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2013-2014 Roman Shtylman <shtylman+expressjs@gmail.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2012 The Obvious Corporation.
http://obvious.com/

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-------------------------------------------------------------------------
Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright 2010, 2011, Chris Winberry <chris@winberry.net>. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
The ISC License

Copyright (c) Isaac Z. Schlueter, Ben Noordhuis, and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

(MIT)

Copyright (c) 2013 Julian Gruber &lt;julian@juliangruber.com&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Node.js is licensed for use as follows:

---

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS
Copyright (c) 2016 Brian Woodward

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2018 Tobias Reich

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Yannick Croissant
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2016 depcheck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Rod Vagg

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013, Joel Feenstra

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the ESQuery nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JOEL FEENSTRA BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2014-2017 browserify-aes contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----

This library bundles a version of the `fs.realpath` and `fs.realpathSync` methods from Node.js v0.10 under the terms of the Node.js MIT license.

Node's license follows, also included at the header of `old.js` which contains the licensed code:

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agree to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License. 
(The MIT License)

Copyright (c) 2013 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining 
a copy of this software and associated documentation files (the 
'Software'), to deal in the Software without restriction, including 
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. 
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY 
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, 
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE 
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. 
MIT License

Copyright (c) 2012 Airbnb

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all 
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2016 Thomas Watson Steen

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

MIT License

Copyright (c) 2017 crypto-browserify

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Jeremiah Senkpiel <fishrock123@rocketmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2013 Roman Shtylman <shtylman@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (C) 2011 VMware, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# This is the MIT license

Copyright (c) 2015 Joyent Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jameson Little

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.
MIT License

Copyright (c) 2017 Ilya Kantor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2017, Rebecca Turner <me@rebecca.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (C) 2011 by Roly Fentanes

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) Elan Shanker and Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2007-2014, Alexandru Marasteanu <hello [at] alexei [dot] ro>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Aria Minaei

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2017 Calvin Metcalf & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2014-2015 Calvin Metcalf & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

MIT

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the
following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)
Copyright (c) 2012-2015, JP Richardson <jprichardson@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files
(the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify,
merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
 ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

BSD 2-Clause License

Copyright (c) 2018, Tadeusz azurski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)
Copyright (c) 2015 Steven Chim

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2015 John Hiesey

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Copyright (c) 2018 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

=====

`String.fromCharCode` by Mathias Bynens used according to terms of MIT License, as follows:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
ISC License (ISC)

Copyright (c) 2015, Glen Maddern

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
The ISC License

Copyright (c) 2015 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Copyright (c) Ben Briggs <beneb.info@gmail.com> (http://beneb.info)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
The MIT License

Copyright (c) 2013 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013-2018 sha.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 1998 - 2009, Paul Johnston & Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this
list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

Neither the name of the author nor the names of its contributors may be used to
endorse or promote products derived from this software without specific prior
written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2017 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright JS Foundation and other contributors, https://js.foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
# Copyright (c) 2015 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.**

Copyright (c), Mapbox
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of node-pre-gyp nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014, Ryan Zimmerman <opensrc@ryanzim.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 KARASZI Istvn
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh, and other contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2018 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Nadav Ivgi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016-2017, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Copyright (c) 2016-2018 Ari Porad

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The MIT License (MIT)

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2008 Fair Oaks Labs, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)
Copyright (c) 2015 Matt DesLauriers

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER DEALINGS IN THE SOFTWARE.
validate-npm-package-license

Give me a string and I'll tell you if it's a valid npm package license string.

```javascript
var valid = require('validate-npm-package-license');
```

SPDX license identifiers are valid license strings:

```javascript
var assert = require('assert');
var validSPDXExpression = {
validForNewPackages: true,
```
validForOldPackages: true,
spdx: true
};

assert.deepEqual(valid('MIT'), validSPDXExpression);
assert.deepEqual(valid('BSD-2-Clause'), validSPDXExpression);
assert.deepEqual(valid('Apache-2.0'), validSPDXExpression);
assert.deepEqual(valid('ISC'), validSPDXExpression);
```

The function will return a warning and suggestion for nearly-correct license identifiers:
```
```javascript
assert.deepEqual(
valid('Apache 2.0'),
{
validForOldPackages: false,
validForNewPackages: false,
warnings: [
'license should be ' +
'a valid SPDX license expression (without "LicenseRef"), ' +
'"UNLICENSED", or ' +
'"SEE LICENSE IN <filename>"',
'license is similar to the valid expression "Apache-2.0"'
]
}
);
```javascript

SPDX expressions are valid, too ...
```
```javascript
// Simple SPDX license expression for dual licensing
assert.deepEqual(
valid('GPL-3.0-only OR BSD-2-Clause'),
validSPDXExpression
);
```javascript

... except if they contain `LicenseRef`:
```
```javascript
var warningAboutLicenseRef = {
validForOldPackages: false,
validForNewPackages: false,
spdx: true,
warnings: [
'license should be ' +
'a valid SPDX license expression (without "LicenseRef"), ' +
```
If you can't describe your licensing terms with standardized SPDX identifiers, put the terms in a file in the package and point users there:

```javascript
assert.deepEqual(
  valid('SEE LICENSE IN LICENSE.txt'),
  {
    validForNewPackages: true,
    validForOldPackages: true,
    inFile: 'LICENSE.txt'
  }
);
assert.deepEqual(
  valid('SEE LICENSE IN license.md'),
  {
    validForNewPackages: true,
    validForOldPackages: true,
    inFile: 'license.md'
  }
);
```

If there aren't any licensing terms, use `UNLICENSED`:

```javascript
var unlicensed = {
  validForNewPackages: true,
  validForOldPackages: true,
  unlicensed: true
};
assert.deepEqual(valid('UNLICENSED'), unlicensed);
```
assert.deepEqual(valid('UNLICENCED'), unlicensed);
```

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.474 @babel/plugin-transform-function-name 7.16.0

1.474.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
1.475 spawn-wrap 2.0.0

1.475.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.476 icu 63

1.476.1 Available under license:

## -*-makefile-*-
## Copyright (C) 2016 and later: Unicode, Inc. and others.
## License & terms of use: http://www.unicode.org/copyright.html
## BSD-specific setup (FreeBSD, OpenBSD, NetBSD, *BSD)
## Copyright (c) 1999-2013, International Business Machines Corporation and ## others. All Rights Reserved.

## Commands to generate dependency files
GEN_DEPS.c=$(CC) -E -MM $(DEFS) $(CPPFLAGS)
GEN_DEPS.cc=$(CXX) -E -MM $(DEFS) $(CPPFLAGS)

## Flags for position independent code
SHAREDLIBCFLAGS = -fPIC
SHAREDLIBCXXFLAGS = -fPIC
SHAREDLIBCPPFLAGS = -DPIC

## Additional flags when building libraries and with threads
THREADSCPPFLAGS = -D_REENTRANT
LIBCPPFLAGS =
## Compiler switch to embed a runtime search path
LD_RPATH=
LD_RPATH_PRE=-Wl,-rpath,

## Compiler switch to embed a library name
LD_SONAME = -Wl,-soname -Wl,$(notdir $(MIDDLE_SO_TARGET))

## Shared library options
LD_SOOPTIONS=-Wl,-Bsymbolic

## Shared object suffix
SO = so
## Non-shared intermediate object suffix
STATIC_O = ao

## Compilation rules
%.$(STATIC_O): $(srcdir)/%.c
$(COMPILE.c) $(STATICCPPFLAGS) $(STATICCFLAGS) -o $@ $<
%.$(STATIC_O): $(srcdir)/%.cpp
$(COMPILE.cc) $(STATICCPPFLAGS) $(STATICCXXFLAGS) -o $@ $<
%.$(STATIC_O): $(srcdir)/%.c
$(COMPILE.c) $(DYNAMICCPPFLAGS) $(DYNAMICCFLAGS) -o $@ $<
%.$(STATIC_O): $(srcdir)/%.cpp
$(COMPILE.cc) $(DYNAMICCPPFLAGS) $(DYNAMICCXXFLAGS) -o $@ $<

## Dependency rules
%.d: $(srcdir)/%.c
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.c\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"
%.d: $(srcdir)/%.cpp
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.cc\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"

## Versioned libraries rules
%.d: $(srcdir)/%.c
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.c\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"
%.d: $(srcdir)/%.cpp
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.cc\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"

## Versioned libraries rules
%.d: $(srcdir)/%.c
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.c\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"
%.d: $(srcdir)/%.cpp
@echo "generating dependency information for $<"
@$(SHELL) -ec "$\(GEN\_DEPS.cc\) $< \| sed \"s/\([^<]*\)\(.*(\.o)\)/\1.o $@ : /g\" \| rm -f $@"
## Bind internal references

# LDflags that pkgdata will use
BIR_LDFLAGS= -Wl,-Bsymbolic

# Dependencies [i.e. map files] for the final library
BIR_DEPS=

## Remove shared library 's'
STATIC_PREFIX_WHEN_USED =
STATIC_PREFIX =

## End BSD-specific setup

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2020 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------
Third-Party Software Licenses

This section contains third-party software notices and/or additional terms for licensed third-party software components included within ICU libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.


# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
The BSD License
http://opensource.org/licenses/bsd-license.php
Copyright (C) 2006-2008, Google Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The word list in cjdict.txt are generated by combining three word lists listed below with further processing for compound word breaking. The frequency is generated with an iterative training against Google web corpora.

* Libtabe (Chinese)
  - https://sourceforge.net/project/?group_id=1519
  - Its license terms and conditions are shown below.

* IPADIC (Japanese)
  - http://chasen.aist-nara.ac.jp/chasen/distribution.html
  - Its license terms and conditions are shown below.

---------COPYING.libtabe ---- BEGIN------------------
/*
 * Copyright (c) 1999 TaBE Project.
 * Copyright (c) 1999 Pai-Hsiang Hsiao.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * . Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * . Neither the name of the TaBE Project nor the names of its
 * contributors may be used to endorse or promote products derived
 * from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 */

/*
 * Copyright (c) 1999 Computer Systems and Communication Lab,
 * Institute of Information Science, Academia Sinica. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * . Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
# * distribution.
# * Neither the name of the Computer Systems and Communication Lab
# * nor the names of its contributors may be used to endorse or
# * promote products derived from this software without specific
# * prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
#
# ---------------COPYING.libtable-----END-----------------------------
#
#
# ---------------COPYING.ipadic-----BEGIN-------------------------------
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# to users as so produced on an experimental basis. Accordingly, the
# program is provided without any warranty whatsoever, whether express,
# implied, statutory or otherwise. The term "warranty" used herein
# includes, but is not limited to, any warranty of the quality,
# performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
# Each user of the program will agree and understand, and be deemed to
# have agreed and understood, that there is no warranty whatsoever for
# the program and, accordingly, the entire risk arising from or
# otherwise connected with the program is assumed by the user.
#
# Therefore, neither ICOT, the copyright holder, or any other
# organization that participated in or was otherwise related to the
# development of the program and their respective officials, directors,
# officers and other employees shall be held liable for any and all
# damages, including, without limitation, general, special, incidental
# and consequential damages, arising out of or otherwise in connection
# with the use or inability to use the program or any product, material
# or result produced or otherwise obtained by using the program,
# regardless of whether they have been advised of, or otherwise had
# knowledge of, the possibility of such damages at any time during the
# project or thereafter. Each user will be deemed to have agreed to the
# foregoing by his or her commencement of use of the program. The term
# "use" as used herein includes, but is not limited to, the use,
# modification, copying and distribution of the program and the
# production of secondary products from the program.
#
# In the case where the program, whether in its original form or
# modified, was distributed or delivered to or received by a user from
# any person, organization or entity other than ICOT, unless it makes or
# grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.
#
# ------------------COPYING.ipadic----END-------------------------------

3. Lao Word Break Dictionary Data (laodict.txt)

# Copyright (c) 2013 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
#                (copied below)
#
# This file is derived from the above dictionary, with slight
# modifications.
# ---------------------------------------------------------------
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification,
# are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in
# binary form must reproduce the above copyright notice, this list of
# conditions and the following disclaimer in the documentation and/or
# other materials provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
# ------------------------------------------------------------------------------------------------
4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
# github.com/kanyawtech/myanmar-karen-word-lists
#
# --------------------------------------------------------------------------

# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
# disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided
# with the distribution.
#
# Neither the name Myanmar Karen Word Lists, nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
# BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
# EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
# TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
# DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
# ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
# TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
# THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
# SUCH DAMAGE.
# --------------------------------------------------------------------------

5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone
Database for its time zone support. The ownership of the TZ database
is explained in BCP 175: Procedure for Maintaining the Time Zone
Database section 7.

# 7. Database Ownership
The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

6. Google double-conversion

Copyright 2006-2011, the V8 project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.477 okio 2.8.0
1.477.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/ByteString.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/internal/-Utf8.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/-Util.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/-Platform.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/-Platform.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/internal/RealBufferedSink.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 Square, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/ByteString.kt
jar/commonMain/okio/internal/SegmentedByteString.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/internal/RealBufferedSource.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/BufferedSource.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/RealBufferedSink.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Okio.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/RealBufferedSource.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Source.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Sink.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Buffer.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/internal/Buffer.kt
*/opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Timeout.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/PeekSource.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/Throttler.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUtf8.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingTimeout.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
Open Source Used In DNAC Platform Arica 1866

/*

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2017 Square, Inc.
 *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

*/

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Options.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/Plpe.kt
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2014 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/GzipSource.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/InflaterSource.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingSink.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/jvmMain/okio/BufferedSource.kt
* /opt/cola/permits/1096821398_1601900245.47/0/okio-2-8-0-sources-jar/commonMain/okio/Segment.kt
1.478 buffer-indexof-polyfill 1.0.2 1.0.2

1.478.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Sarosia

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.479 react 16.4.2 16.4.2

1.479.1 Available under license:

MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.480 update-notifier 5.1.0
1.480.1 Available under license :
Copyright Google

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.481 netty-codec 4.1.42.Final
1.481.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/ByteToMessageDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/PrematureChannelClosureException.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/compression/ZlibDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/compression/SnappyFrameEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/compression/DecompressionException.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/serialization/ClassResolvers.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/marshalling/package-info.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/marshalling/ThreadLocalMarshallerProvider.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/marshalling/DefaultMarshallerProvider.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/marshalling/package-info.java
Open Source Used In DNAC Platform Arica 1871

No license file was found, but licenses were detected in source scan.

/ *
/ * Copyright 2015 The Netty Project
/ *
/ * The Netty Project licenses this file to you under the Apache License,
/ * version 2.0 (the "License"); you may not use this file except in compliance
/ * with the License. You may obtain a copy of the License at:
/ *
/ *   http://www.apache.org/licenses/LICENSE-2.0
/ *
/ * Unless required by applicable law or agreed to in writing, software
/ * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
/ * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
/ * License for the specific language governing permissions and limitations
/ * under the License.
/ */

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Snappy.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/MessageToMessageDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/serialization/ObjectDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/ReplayingDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/UnsupportedMessageTypeException.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/TooLongFrameException.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/Delimiters.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/CodecException.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/marshalling/CompatibleMarshallingDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/protobuf/package-info.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/marshalling/ChannelBufferByteInput.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/base64/Base64Decoder.java

Open Source Used In DNAC Platform Arica 1873
jar/io/netty/handler/codec/UnsupportedValueConverter.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32LengthFieldPrepender.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/HeadersUtils.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/ProtocolDetectionResult.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32FrameDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufEncoderNano.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/protobuf/ProtobufDecoderNano.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE
2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/META-INF/maven/io.netty/netty-codec/pom.xml

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/string/LineEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/compression/ByteBufChecksum.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/DatagramPacketDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/string/LineSeparator.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/CodecOutputList.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/DateFormatter.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/compression/CompressionUtil.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
  jar/io/netty/handler/codec/CodecOutputList.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2012 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/*
* Written by Robert Harder and released to the public domain, as explained at
* http://creativecommons.org/licenses/publicdomain
*/
/**
* Utility class for [ByteBuf] that encodes and decodes to and from Base64 notation.

* The encoding and decoding algorithm in this class has been derived from Robert Harder's Public Domain Base64 Encoder/Decoder.

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/base64/Base64.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/AsciiHeadersEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2MTFAndRLE2StageEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2MoveToFrontTable.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Lz4Constants.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/LzfDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/SnappyFramedDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Crc32.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2Decoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Lz4FrameDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2Constants.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/FastLzFrameDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/FastLzFrameSort.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/FastLzFrameEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/DecoderResultProvider.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/LzfEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/LzmaFrameEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2BlockCompressor.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2BlockDecompressor.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2Rand.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2Encoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Lz4FrameEncoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Lz4FrameDecoder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.


No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */


No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* A decoder that splits the received @link ByteBuf}s dynamically by the
* value of the length field in the message. It is particularly useful when you
* decode a binary message which has an integer header field that represents the
* length of the message body or the whole message.
* *
* @link LengthFieldBasedFrameDecoder} has many configuration parameters so
* that it can decode any message with a length field, which is often seen in
* proprietary client-server protocols. Here are some example that will give
* you the basic idea on which option does what.
* *
* <h3>2 bytes length field at offset 0, do not strip header</h3>
* *
* The value of the length field in this example is &lt;tt&gt;12 (0x0C)&lt;/tt&gt; which
* represents the length of "HELLO, WORLD". By default, the decoder assumes
* that the length field represents the number of the bytes that follows the
* length field. Therefore, it can be decoded with the simplistic parameter
* combination.
* <pre>
* <b>lengthFieldOffset</b>   = <b>0</b>
* <b>lengthFieldLength</b>   = <b>2</b>
* lengthAdjustment    = 0
* initialBytesToStrip = 0 (= do not strip header)
*
* BEFORE DECODE (14 bytes)         AFTER DECODE (14 bytes)
* +--------+----------------+      +--------+----------------+
* | Length | Actual Content |----->| Length | Actual Content |
* | 0x000C | "HELLO, WORLD" |      | 0x000C | "HELLO, WORLD" |
* +--------+----------------+      +--------+----------------+
* </pre>
* *
* <h3>2 bytes length field at offset 0, strip header</h3>
* *
* Because we can get the length of the content by calling
* @link ByteBuf#readableBytes(), you might want to strip the length
* field by specifying &lt;tt&gt;initialBytesToStrip&lt;/tt&gt;. In this example, we
* specified &lt;tt&gt;2&lt;/tt&gt;, that is same with the length of the length field, to
* strip the first two bytes.

* <pre>
* lengthFieldOffset = 0
* lengthFieldLength = 2
* lengthAdjustment = 0
* <b>initialBytesToStrip</b> = <b>2</b> (= the length of the Length field)
* 
* BEFORE DECODE (14 bytes) AFTER DECODE (12 bytes)
* +--------+----------------+      +----------------+
* | Length | Actual Content |----->| Actual Content |
* | 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
* +--------+----------------+      +----------------+
* </pre>

* <h3>2 bytes length field at offset 0, do not strip header, the length field
represents the length of the whole message</h3>

* In most cases, the length field represents the length of the message body
only, as shown in the previous examples. However, in some protocols, the
length field represents the length of the whole message, including the
message header. In such a case, we specify a non-zero
<tt>lengthAdjustment</tt>. Because the length value in this example message
is always greater than the body length by <tt>2</tt>, we specify <tt>-2</tt> for compensation.
* <pre>
* lengthFieldOffset = 0
* lengthFieldLength = 2
* <b>lengthAdjustment</b> = <b>-2</b> (= the length of the Length field)
* initialBytesToStrip = 0
* 
* BEFORE DECODE (14 bytes) AFTER DECODE (14 bytes)
* +--------+----------------+      +--------+----------------+
* | Length | Actual Content |----->| Length | Actual Content |
* | 0x000E | "HELLO, WORLD" |      | 0x000E | "HELLO, WORLD" |
* +--------+----------------+      +--------+----------------+
* </pre>

* <h3>3 bytes length field at the end of 5 bytes header, do not strip header</h3>

* The following message is a simple variation of the first example. An extra
header value is prepended to the message. <tt>lengthAdjustment</tt> is zero
again because the decoder always takes the length of the prepended data into
account during frame length calculation.
* <pre>
* lengthFieldOffset = 0
* lengthFieldLength = 2
* <b>lengthAdjustment</b> = <b>-2</b> (= the length of Header 1)
* 
* BEFORE DECODE (14 bytes) AFTER DECODE (14 bytes)
* +--------+----------------+      +--------+----------------+
* | Length | Actual Content |----->| Length | Actual Content |
* | 0x000E | "HELLO, WORLD" |      | 0x000E | "HELLO, WORLD" |
* +--------+----------------+      +--------+----------------+
* </pre>

* <h3>3 bytes length field at the end of 5 bytes header, do not strip header</h3>

* The following message is a simple variation of the first example. An extra
header value is prepended to the message. <tt>lengthAdjustment</tt> is zero
again because the decoder always takes the length of the prepended data into
account during frame length calculation.
* BEFORE DECODE (17 bytes) AFTER DECODE (17 bytes)
* +----------------------------+----------------------------+
* | Header 1 | Length | Actual Content |----->| Header 1 | Length | Actual Content |
* | 0xCAFE | 0x00000C | "HELLO, WORLD" |      | 0xCAFE | 0x00000C | "HELLO, WORLD" |
* +----------------------------+----------------------------+
* </pre>
*
* BEFORE DECODE (17 bytes) AFTER DECODE (17 bytes)
* +----------------------------+----------------------------+
* | Length | Header 1 | Actual Content |----->| Length | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE | "HELLO, WORLD" |      | 0x00000C | 0xCAFE | "HELLO, WORLD" |
* +----------------------------+----------------------------+
* </pre>
*
* BEFORE DECODE (17 bytes) AFTER DECODE (17 bytes)
* +----------------------------+----------------------------+
* | Length | Header 1 | Actual Content |----->| Length | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE | "HELLO, WORLD" |      | 0x00000C | 0xCAFE | "HELLO, WORLD" |
* +----------------------------+----------------------------+
* </pre>
*
* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +----------------------------+----------------------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +----------------------------+----------------------------+
Let's give another twist to the previous example. The only difference from the previous example is that the length field represents the length of the whole message instead of the message body, just like the third example.

We have to count the length of HDR1 and Length into lengthAdjustment.

Please note that we don't need to take the length of HDR2 into account because the length field already includes the whole header length.

```
lengthFieldOffset = 1
lengthFieldLength = 2
lengthAdjustment = -3 (= the length of HDR1 + LEN, negative)
initialBytesToStrip = 3
```

**BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)**

```
+--------------+-------+----------------+    +----------------------+
| HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
| 0xCA | 0x0010 | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
+--------------+-------+----------------+
```

@see LengthFieldPrepender

```
*

Found in path(s):

/opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/DefaultHeadersImpl.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/CharSequenceValueConverter.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/ValueConverter.java

No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2013 The Netty Project
*  
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*  
* http://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/xml/XmlFrameDecoder.java
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-jar/io/netty/handler/codec/compression/Crc32c.java

No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2012 The Netty Project
*  
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*  
* http://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
/*
/*  * Written by Robert Harder and released to the public domain, as explained at
/*  * http://creativecommons.org/licenses/publicdomain
/*
/*  * Enumeration of supported Base64 dialects.
/*  * <p>
/*  * The internal lookup tables in this class has been derived from
/*  * <a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain
/*  * Base64 Encoder/Decoder</a>.
/*
*/

Found in path(s):
* /opt/cola/permits/1144317898_1615901243.38/0/netty-codec-4-1-42-final-sources-3-
jar/io/netty/handler/codec/base64/Base64Dialect.java

1.482 js-tokens 4.0.0
1.482.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017, 2018 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.483 lucene-analyzers-common 7.4.0
1.483.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from unicode conversion examples available at
http://www.unicode.org/Public/PROGRAMS/CVTUTF. Here is the copyright
from those sources:

/*
 * Copyright 2001-2004 Unicode, Inc.
 *
 * Disclaimer
 *
 * This source code is provided as is by Unicode, Inc. No claims are
 * made as to fitness for any particular purpose. No warranties of any
 * kind are expressed or implied. The recipient agrees to determine
 * applicability of information provided. If this file has been
 * purchased on magnetic or optical media from Unicode, Inc., the
 * sole remedy for any claim will be exchange of defective media
 * within 90 days of receipt.
 *
 * Limitations on Rights to Redistribute This Code
 *
 * Unicode, Inc. hereby grants the right to freely use the information
 * supplied in this file in the creation of products supporting the
 * Unicode Standard, and to make copies of this file in any form
 * for internal or external distribution as long as this notice
 * remains attached.
 */

Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was
derived from Python 2.4.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/
Some code in `core/src/java/org/apache/lucene/util/UnicodeUtil.java` was
derived from Python 3.1.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in `core/src/java/org/apache/lucene/util/automaton` was
derived from Brics automaton sources available at
www.brics.dk/automaton/. Here is the copyright from those sources:

/*
 * Copyright (c) 2001-2009 Anders Moeller
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

The levenshtein automata tables in `core/src/java/org/apache/lucene/util/automaton`
were automatically generated with the moman/finenight FSA package.
Here is the copyright for those sources:

# Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
# Permission is hereby granted, free of charge, to any person
# obtaining a copy of this software and associated documentation
# files (the “Software”), to deal in the Software without
# restriction, including without limitation the rights to use,
# copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following
# conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
# OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
# OTHER DEALINGS IN THE SOFTWARE.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
 * Copyright (C) 1999-2010, International Business Machines
 * Corporation and others. All Rights Reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, and/or sell copies of the
 * Software, and to permit persons to whom the Software is furnished to do so,
 * provided that the above copyright notice(s) and this permission notice appear
 * in all copies of the Software and that both the above copyright notice(s) and
 * this permission notice appear in supporting documentation.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
 * IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
 * LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
 * ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
 * IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
 * OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
 *
 * Except as contained in this notice, the name of a copyright holder shall not
 * be used in advertising or otherwise to promote the sale, use or other
 * dealings in this Software without prior written authorization of the
 * copyright holder.
 */
The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names "Center for Intelligent Information Retrieval" and
"University of Massachusetts” must not be used to endorse or promote products derived from this software without prior written permission. To obtain permission, contact info@ciir.cs.umass.edu.

This software is provided by University of Massachusetts and other contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright holders or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright holder or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

---
The dictionary comes from Morfologik project. Morfologik uses data from Polish ispell/myspell dictionary hosted at http://www.sjp.pl/slownik/en/ and is licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike. The part-of-speech tags were added in Morfologik project and are not found in the data from sjp.pl. The tagset is similar to IPI PAN tagset.

---

The following license applies to the Morfeusz project, used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/

Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski, Marcin Woliski, Robert Woosz

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY COPYRIGHT HOLDERS AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects, including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such as the Unicode Character Database. See http://unicode.org/copyright.html for more details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were automatically generated with the moman/finenight FSA library, created by Jean-Philippe Barrette-LaPierre. This library is available under an MIT license, see http://sites.google.com/site/rrettesite/moman and http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0. See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0 See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin g Package (jaspell): http://jaspell.sourceforge.net/ License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in analysis/common/src/java/net/sf/snowball were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in analysis/common/src/resources/org/apache/lucene/analysis/snowball were developed by Martin Porter and Richard Boulton.
The full snowball package is available from http://snowball.tartarus.org/

The KStem stemmer in
analysis/common/src/org/apache/lucene/analysis/en was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst) under the BSD-license.

The Arabic, Persian, Romanian, Bulgarian, Hindi and Bengali analyzers (common) come with a default stopword list that is BSD-licensed created by Jacques Savoy. These files reside in:

- analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt
- analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt
- analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt
- analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt
- analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt
- analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt

See [http://members.unine.ch/jacques.savoy/clef/index.html](http://members.unine.ch/jacques.savoy/clef/index.html).

The German, Spanish, Finnish, French, Hungarian, Italian, Portuguese, Russian and Swedish light stemmers (common) are based on BSD-licensed reference implementations created by Jacques Savoy and Ljiljana Dolamic. These files reside in:

- analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/fi/FinnishLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
- analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed by the Egothor project [http://egothor.sf.net/](http://egothor.sf.net/), created by Leo Galambos, Martin Kvapil, and Edmond Nolan.

The Polish analyzer (stempel) comes with a default stopword list that is BSD-licensed created by the Carrot2 project. The file resides in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt. See [http://project.carrot2.org/license.html](http://project.carrot2.org/license.html).

The SmartChineseAnalyzer source code (smartcn) was provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/) is derived from Unicode data such as the Unicode Character Database. See [http://unicode.org/copyright.html](http://unicode.org/copyright.html) for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software developed by Dawid Weiss and Marcin Mikowski ([http://morfologik.blogspot.com/](http://morfologik.blogspot.com/)).
Morfologik uses data from Polish ispell/myspell dictionary
(http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia)
LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP)
(http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original
source code for this can be found at http://www.eclipse.org/jetty/downloads.php

===========================================================================
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

===========================================================================
mecab-ipadic-2.7.0-20070801 Notice
===========================================================================

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise contravene any of the laws and regulations of the countries having jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to have agreed and understood, that there is no warranty whatsoever for the program and, accordingly, the entire risk arising from or otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other organization that participated in or was otherwise related to the development of the program and their respective officials, directors, officers and other employees shall be held liable for any and all damages, including, without limitation, general, special, incidental and consequential damages, arising out of or otherwise in connection with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

===========================================================================
Nori Korean Morphological Analyzer - Apache Lucene Integration
===========================================================================

Open Source Used In DNAC Platform Arica 1898
This software includes a binary and/or source version of data from

mecab-ko-dic-2.0.3-20170922

which can be obtained from

https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz

1.484 jaxb-runtime 2.3.1

1.485 classgraph 4.6.32

1.486 netty-handler 4.1.42.Final

1.486.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):

* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/traffic/AbstractTrafficShapingHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.
jar/io/netty/handler/traffic/GlobalChannelTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/traffic/GlobalTrafficShapingHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/traffic/TrafficCounter.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/traffic/package-info.java
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/META-INF/native-image/io.netty/handler/native-image.properties
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/AbstractSniHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/SslCompletionEvent.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/SniCompletionEvent.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/Java9SslEngine.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/Java9SslUtils.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/OcspClientHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/OptionalSslHandler.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
jar/io/netty/handler/stream/ChunkedNioFile.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/META-INF/maven/io.netty/netty-handler/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/SslMasterKeyHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/address/DynamicAddressConnectHandler.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/address/package-info.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/OpenSslPrivateKeyMethod.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// Try the OpenJDK's proprietary implementation.

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
  jar/io/netty/handler/ssl/Util/SelfSignedCertificate.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
  jar/io/netty/handler/ssl/Util/UniqueIpFilter.java

Open Source Used In DNAC Platform Arica 1905
jar/io/netty/handler/ssl/OpenSslServerSessionContext.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/flow/FlowControlHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/OpenSslKeyMaterialManager.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/flush/package-info.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/PemEncoded.java
* /opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-
jar/io/netty/handler/ssl/OpenSslCertificateException.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/PemValue.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/FlushConsolidationHandler.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/ReferenceCountedOpenSslContext.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/PemX509Certificate.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/PemPrivateKey.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/Java8SslUtils.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/DelegatingSslContext.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/ReferenceCountedOpenSslEngine.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/util/X509TrustManagerWrapper.java
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/flow/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
*/opt/cola/permits/1144317995_1615901296.45/0/netty-handler-4-1-42-final-sources-3-jar/io/netty/handler/ssl/SslHandshakeCompletionEvent.java

No license file was found, but licenses were detected in source scan.
1.487.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.488 has-ansi 2.0.0 2.0.0

1.488.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.489 es-abstract 1.17.7

1.489.1 Available under license:

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.490 drill-format-esri 1.19.0

1.490.1 Available under license:

Drill : Contrib : Format : Esri
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.491 lodash.isstring 4.0.1 4.0.1

1.491.1 Available under license:

Copyright 2012-2016 The Dojo Foundation <http://dojofoundation.org/>
Based on Underscore.js, copyright 2009-2016 Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.492 amqplib 0.5.6 0.5.6

1.492.1 Available under license:

amqplib copyright (c) 2013, 2014
Michael Bridgen <mikeb@squaremobius.net>

This package, "amqplib", is licensed under the MIT License. A copy may
be found in the file LICENSE-MIT in this directory, or downloaded from
http://opensource.org/licenses/MIT.
The MIT License (MIT)

Copyright (c) 2013, 2014 Michael Bridgen <mikeb@squaremobius.net>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.493 forwarded 0.1.2 0.1.2

1.493.1 Available under license :
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.494 home-or-tmp 2.0.0

1.494.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.495 mkdirp 0.5.5 0.5.5

1.495.1 Available under license :
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.496 snake-yaml 1.26
1.496 Available under license:

Found license 'GNU Lesser General Public License' in 'This module is multi-licensed and may be used under the terms EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html AL, Apache License, V2.0 or later, http://www.apache.org/licenses BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

Found license 'Eclipse Public License 1.0' in 'This module is multi-licensed and may be used under the terms EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html AL, Apache License, V2.0 or later, http://www.apache.org/licenses BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

1.497 regenerate 1.4.2

1.497.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.498 json-schema-traverse 0.3.1 0.3.1

1.498.1 Available under license:

MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.499 ansi-regex 3.0.0 3.0.0
1.499.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.500 jsreport-version-control 1.6.0 1.6.0
1.500.1 Available under license :
MIT License

Copyright (c) 2017 jsreport
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.501 jffi 1.2.16
1.501.1 Available under license :
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form,
including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
1.502 fs 0.0.2 0.0.2

1.502.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.503 escodegen 1.8.1

1.503.1 Available under license:

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.504 supports-color 5.5.0
1.504.1 Available under license:
  MIT

1.505 jsreport 2.11.0 2.11.0
1.505.1 Available under license:
  GNU LESSER GENERAL PUBLIC LICENSE
  Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:
a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to recombinne or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombinning or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.506 loose-envify 1.4.0 1.4.0

1.506.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Andres Suarez <zertos@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.507 chrome-page-eval 1.3.0 1.3.0
1.507.1 Available under license:
The MIT License (MIT)

Copyright (c) 2018 bjrmatos

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.508 ant-junit 1.10.9

1.508.1 Available under license:

/*
 *     Apache License
 *     Version 2.0, January 2004
 *     http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction,
 * and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by
 * the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all
 * other entities that control, are controlled by, or are under common
 * control with that entity. For the purposes of this definition,
 * "control" means (i) the power, direct or indirect, to cause the
 * direction or management of such entity, whether by contract or
 * otherwise, or (ii) ownership of fifty percent (50%) or more of the

Open Source Used In DNAC Platform Arica 1931
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* copyright license to reproduce, prepare Derivative Works of,
* publicly display, publicly perform, sublicense, and distribute the
* Work and such Derivative Works in Source or Object form.
*
* 3. Grant of Patent License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* (except as stated in this section) patent license to make, have made,
* use, offer to sell, sell, import, and otherwise transfer the Work,
* where such license applies only to those patent claims licensable
* by such Contributor that are necessarily infringed by their
* Contribution(s) alone or by combination of their Contribution(s)
* with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.
*
* 4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:
*
* (a) You must give any other recipients of the Work or
* Derivative Works a copy of this License; and
*
* (b) You must cause any modified files to carry prominent notices
* stating that You changed the files; and
*
* (c) You must retain, in the Source form of any Derivative Works
* that You distribute, all copyright, patent, trademark, and
* attribution notices from the Source form of the Work,
* excluding those notices that do not pertain to any part of
* the Derivative Works; and
*
* (d) If the Work includes a "NOTICE" text file as part of its
* distribution, then any Derivative Works that You distribute must
* include a readable copy of the attribution notices contained
* within such NOTICE file, excluding those notices that do not
* pertain to any part of the Derivative Works, in at least one
* of the following places: within a NOTICE text file distributed
* as part of the Derivative Works; within the Source form or
* documentation, if provided along with the Derivative Works; or,
* within a display generated by the Derivative Works, if and
* wherever such third-party notices normally appear. The contents
* of the NOTICE file are for informational purposes only and
* do not modify the License. You may add Your own attribution
* notices within Derivative Works that You distribute, alongside
* or as an addendum to the NOTICE text from the Work, provided
* that such additional attribution notices cannot be construed
* as modifying the License.

* You may add Your own copyright statement to Your modifications and
* may provide additional or different license terms and conditions
* for use, reproduction, or distribution of Your modifications, or
* for any such Derivative Works as a whole, provided Your use,
* reproduction, and distribution of the Work otherwise complies with
* the conditions stated in this License.

* 5. Submission of Contributions. Unless You explicitly state otherwise,
* any Contribution intentionally submitted for inclusion in the Work
* by You to the Licensor shall be under the terms and conditions of
* this License, without any additional terms or conditions.
* Notwithstanding the above, nothing herein shall supersede or modify
* the terms of any separate license agreement you may have executed
* with Licensor regarding such Contributions.

* 6. Trademarks. This License does not grant permission to use the trade
* names, trademarks, service marks, or product names of the Licensor,
* except as required for reasonable and customary use in describing the
* origin of the Work and reproducing the content of the NOTICE file.

* 7. Disclaimer of Warranty. Unless required by applicable law or
* agreed to in writing, Licensor provides the Work (and each
* Contributor provides its Contributions) on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
* implied, including, without limitation, any warranties or conditions
* of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
* PARTICULAR PURPOSE. You are solely responsible for determining the
* appropriateness of using or redistributing the Work and assume any
* risks associated with Your exercise of permissions under this License.

* 8. Limitation of Liability. In no event and under no legal theory,
* whether in tort (including negligence), contract, or otherwise,
* unless required by applicable law (such as deliberate and grossly
* negligent acts) or agreed to in writing, shall any Contributor be
* liable to You for damages, including any direct, indirect, special,
* incidental, or consequential damages of any character arising as a
* result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill,
* work stoppage, computer failure or malfunction, or any and all
* other commercial damages or losses), even if such Contributor
* has been advised of the possibility of such damages.
* 9. Accepting Warranty or Additional Liability. While redistributing
* the Work or Derivative Works thereof, You may choose to offer,
* and charge a fee for, acceptance of support, warranty, indemnity,
* or other liability obligations and/or rights consistent with this
* License. However, in accepting such obligations, You may act only
* on Your own behalf and on Your sole responsibility, not on behalf
* of any other Contributor, and only if You agree to indemnify,
* defend, and hold each Contributor harmless for any liability
* incurred by, or claims asserted against, such Contributor by reason
* of your accepting any such warranty or additional liability.
*
* END OF TERMS AND CONDITIONS
*
* APPENDIX: How to apply the Apache License to your work.
*
* To apply the Apache License to your work, attach the following
* boilerplate notice, with the fields enclosed by brackets "[]"
* replaced with your own identifying information. (Don't include
* the brackets!) The text should be enclosed in the appropriate
* comment syntax for the file format. We also recommend that a
* file or class name and description of purpose be included on the
* same "printed page" as the copyright notice for easier
* identification within third-party archives.
*
* Copyright [yyyy] [name of copyright owner]
*
* Licensed under the Apache License, Version 2.0 (the "License);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*      https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

W3C SOFTWARE NOTICE AND LICENSE
https://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other
related items) is being provided by the copyright holders under the following
license. By obtaining, using and/or copying this work, you (the licensee) agree
that you have read, understood, and will comply with the following terms and
conditions.
Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following in ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd
SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
Apache Ant
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

1.509 commons-net 3.6
1.509.1 Available under license :
Apache Commons Net
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.510 babel-plugin-transform-async-generator-functions 6.24.1
1.510.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.511 cli-width 2.2.1 2.2.1

1.511.1 Available under license:
Copyright (c) 2015, Ilya Radchenko <ilya@burstcreations.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.512 form-data 1.0.0-rc4

1.512.1 Available under license:
Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.513 content-type 1.0.4 1.0.4

1.513.1 Available under license:
(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.514 libxtst 1.2.2-1

1.514.1 Available under license:
Copyright 1990, 1991 by UniSoft Group Limited

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

******************************************************************************

Copyright 1995 Network Computing Devices

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Network Computing Devices not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

NETWORK COMPUTING DEVICES DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NETWORK COMPUTING DEVICES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

******************************************************************************

Copyright 2005 Red Hat, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Red Hat not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Red Hat makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

RED HAT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL RED HAT BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
PERFORMANCE OF THIS SOFTWARE.

***************************************************************************

Copyright 1992 by UniSoft Group Ltd.

Permission to use, copy, modify, and distribute this documentation for any purpose and without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. UniSoft makes no representations about the suitability for any purpose of the information in this document. This documentation is provided "as is" without express or implied warranty.

***************************************************************************

Copyright 1992, 1994, 1995 X Consortium

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

***************************************************************************

Copyright 1994 Network Computing Devices, Inc.

Permission to use, copy, modify, distribute, and sell this documentation for any purpose is hereby granted without fee, provided that the above copyright notice and this permission notice appear in all copies. Network Computing Devices, Inc.
makes no representations about the suitability for any purpose of the information in this document. This documentation is provided "as is" without express or implied warranty.

1.515 yaml 1.10.2 1.10.2
1.515.1 Available under license:
Copyright 2018 Eemeli Aro <eemeli@gmail.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.516 formidable 1.2.1
1.516.1 Available under license:
Copyright (C) 2011 Felix Geisendrfer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.517 unbox-primitive 1.0.1
1.517.1 Available under license:
MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.518 calcite-avatica 1.17.0
1.518.1 Available under license:
Copyright (c) 2013 Coby Chapple
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) Nicolas Gallagher and Jonathan Neal

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Calcite -- Avatica
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Copyright 2008 Google Inc. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

______________________________________________________________

APACHE CALCITE AVATICA SUBCOMPONENTS:
The Apache Calcite Avatica project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

-----------------------------------------------------------------------
The MIT License
-----------------------------------------------------------------------

The Apache Calcite project bundles the following files under the MIT License:

- site
  Parts of the web site generated by Jekyll (http://jekyllrb.com/)
  Copyright (c) 2008-2015 Tom Preston-Werner
- site/_sass/_font-awesome.scss
  Font-awesome css files v4.1.0 (http://fortawesome.github.io/Font-Awesome/)
  Copyright (c) 2013  Dave Gandy
- site/_sass/_normalize.scss
  normalize.css v3.0.2 | git.io/normalize
  Copyright (c) Nicolas Gallagher and Jonathan Neal
- site/_sass/_gridism.scss
  Gridism: A simple, responsive, and handy CSS grid by @cobyism
  https://github.com/cobyism/gridism
  Copyright (c) 2013 Coby Chapple
- site/js/html5shiv.min.js
  HTML5 Shiv 3.7.2 | @afarkas @jdalton @jon_neal @rem
- site/js/respond.min.js
  Respond.js v1.4.2: min/max-width media query polyfill
  Copyright 2013 Scott Jehl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The Open Font License

The Apache Calcite project bundles the following fonts under the
SIL Open Font License (OFL) - http://scripts.sil.org/OFL/

- site/fonts/fontawesome-webfont.*
  Font-awesome font files v4.0.3 (http://fortawesome.github.io/Font-Awesome/)

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses...
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

---------------------------------------------------------

APACHE CALCITE AVATICA SUBCOMPONENTS:

The Apache Calcite Avatica project contains subcomponents with separate copyright
notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following
licenses.

---------------------------------------------------------

3-clause BSD license

---------------------------------------------------------

The Apache Calcite Avatica project bundles HSQLDB, which is available
under the following "3-clause BSD" license:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
 * For work developed by the HSQL Development Group:
 *
 * Copyright (c) 2001-2018, The HSQL Development Group
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
 *
 * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 *
 * Neither the name of the HSQL Development Group nor the names of its
 * contributors may be used to endorse or promote products derived from this
 * software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG,
 * OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* For work originally developed by the Hypersonic SQL Group:
* 
* Copyright (c) 1995-2000, The Hypersonic SQL Group.
* All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
* 
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
* 
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
* 
* Neither the name of the Hypersonic SQL Group nor the names of its
* contributors may be used to endorse or promote products derived from this
* software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* This software consists of voluntary contributions made by many individuals
* on behalf of the Hypersonic SQL Group.
*/

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to you under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# Database of contributors to Apache Calcite.
# Pages such as developer.md use this data.
- name: Alan Gates
  apacheId: gates
  githubId: alanfgates
  org: Hortonworks
  role: PMC
- name: Aman Sinha
  apacheId: amansinha
  githubId: amansinha100
  org: MapR
  role: PMC
- name: Ashutosh Chauhan
  apacheId: hashutosh
  githubId: ashutoshc
  org: Hortonworks
  role: PMC
- name: Francis Chuang
  apacheId: francischuang
  githubId: F21
  org: Boostport
  role: Committer
- name: James R. Taylor
  apacheId: jamestaylor
  githubId: JamesRTaylor
  org: Salesforce
  role: PMC
- name: Jacques Nadeau
  apacheId: jacques
  githubId: jacques-n
  org: Dremio
  role: PMC
- name: Jess Camacho Rodriguez
  apacheId: jcamacho
  githubId: jcamachor
  org: Hortonworks
  role: PMC Chair
- name: Jinfeng Ni
apacheId: jni
githubId: jinfengni
org: MapR
role: PMC
- name: John Pullokkaran
  apacheId: jpullokk
  githubId: jpullokkaran
  org: Hortonworks
  role: PMC
- name: Josh Elser
  apacheId: elserj
  githubId: joshelser
  org: Hortonworks
  role: PMC
- name: Julian Hyde
  apacheId: jhyde
  githubId: julianhyde
  org: Hortonworks
  role: PMC
  homepage: http://people.apache.org/~jhyde
- name: Maryann Xue
  apacheId: maryannxue
  githubId: maryannxue
  org: Intel
  role: Committer
- name: Michael Mior
  apacheId: mmior
  githubId: michaelmior
  org: University of Waterloo
  role: Committer
  homepage: http://michael.mior.ca/
- name: Milinda Pathirage
  apacheId: milinda
  githubId: milinda
  org: Indiana University
  role: Committer
  homepage: http://milinda.pathirage.org/
- name: MinJi Kim
  apacheId: minji
  githubId: minji-kim
  org: Dremio
  role: Committer
  avatar: http://web.mit.edu/minjikim/www/minji.png
  homepage: http://web.mit.edu/minjikim/www/
- name: Nick Dimiduk
  apacheId: ndimiduk
  githubId: ndimiduk
  org: Hortonworks
role: PMC
homepage: http://www.n10k.com
- name: Steven Noels
  apacheId: stevenn
  githubId: stevenn
  org: NGData
  role: PMC
- name: Ted Dunning
  apacheId: tdunning
  githubId: tdunning
  org: MapR
  role: PMC
  avatar: https://www.mapr.com/sites/default/files/otherpageimages/ted-circle-80.png
- name: Vladimir Sitnikov
  apacheId: vladimirsitnikov
  githubId: vlsi
  org: NetCracker
  role: PMC
# End contributors.yml

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions.
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

https://fortawesome.github.io/Font-Awesome/license/

Font License

Applies to all desktop and webfont files in the following directory: font-awesome/fonts/.
License: SIL OFL 1.1
URL: http://scripts.sil.org/OFL

Code License

Applies to all CSS and LESS files in the following directories: font-awesome/css/, font-awesome/less/, and font-awesome/scss/.
License: MIT License
URL: http://opensource.org/licenses/mit-license.html

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
APACHE CALCITE AVATICA SUBCOMPONENTS:

The Apache Calcite Avatica project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

3-clause BSD license

The Apache Calcite Avatica project bundles Protocol Buffers, which is available under the following "3-clause BSD" license:

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
/* Copyright (c) 2001-2018, The HSQL Development Group
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 * *
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
 * *
 * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 * *
 * Neither the name of the HSQL Development Group nor the names of its
 * contributors may be used to endorse or promote products derived from this
 * software without specific prior written permission.
 * *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG,
 * OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */
The MIT License (MIT)

Copyright (c) 2014 Alexander Farkas (aFarkas)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2008-present Tom Preston-Werner and Jekyll contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.519 micromatch 3.1.10 3.1.10
1.519.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.520 tough-cookie 2.5.0 2.5.0

1.520.1 Available under license :
Copyright (c) 2015, Salesforce.com, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of Salesforce.com nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.521 swagger-integration 2.0.7

1.521.1 Available under license :
Copyright 2018 SmartBear Software
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at [apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.522 @babel/plugin-proposal-logical-assignment-operators 7.16.0

1.522.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.523 http-signature 1.2.0 1.2.0

1.523.1 Available under license :
Copyright Joyent, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.524 byte-buddy 1.9.3

1.524.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 - 2018 Rafael Winterhalter
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/description/field/FieldDescription.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/dynamic/scaffold/inline/RebaseDynamicTypeBuilder.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/implementation/MethodCall.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/implementation/auxiliary>TypeProxy.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/description/NamedElement.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy/dynamic/Transformer.java
Open Source Used In DNAC Platform Arica 1984

* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/LoadedTypeInitializer.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/bytecode/assign/primitive/PrimitiveWideningDelegate.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/pool/package-info.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/loading/ClassReloadingStrategy.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/loading/ClassLoadingStrategy.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/loading/InjectionClassLoader.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/description/enumeration/package-info.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/bind/annotation/This.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/description/annotation/AnnotationValue.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/matcher/DeclaringAnnotationMatcher.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/bind/annotation/BindingPriority.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/scaffold/inline/AbstractInliningDynamicTypeBuilder.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/description/modifier/SynchronizationState.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/scaffold/subclass/package-info.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/bind/annotation/package-info.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/matcher/AnnotationTypeMatcher.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/dynamic/loading/MultipleParentClassLoader.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/auxiliary/MethodCallProxy.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/bytecode/StackManipulation.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/description/annotation/AnnotationDescription.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/InvocationHandlerAdapter.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/implementation/attribute/TypeAttributeAppender.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/utility/privilege/package-info.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1.9.3-sources.jar/net/bytebuddy/matcher/MethodOverrideMatcher.java
2011 INRIA, France Telecom
* All rights reserved.

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
//  documentation and/or other materials provided with the distribution.
//  3. Neither the name of the copyright holders nor the names of its
//  this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Opcodes.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/CurrentFrame.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Attribute.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/ClassTooLargeException.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/ByteVector.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Handle.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/SymbolTable.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Type.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/FieldVisitor.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/SignatureReader.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/AnnotationWriter.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Context.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Label.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/SignatureWriter.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/TypePath.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/AnnotationVisitor.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Symbol.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/SignatureVisitor.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Frame.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/TypeReference.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/ConstantDynamic.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/MethodTooLargeException.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/Edge.java
* /opt/cola/permits/1258286336_1643052509.41/0/byte-buddy-1-9-3-sources-jar/net/bytebuddy.jar/asm/MethodWriter.java
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-Description: Byte Buddy is a Java library for creating Java classes at run time. This artifact is a build of Byte Buddy with all ASM dependencies repackaged into its own name space.
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-SymbolicName: net.bytebuddy.byte-buddy
Built-By: rafael
Bnd-LastModified: 1540749633545
Bundle-ManifestVersion: 2
Tool: Bnd-3.5.0.201709291849
Main-Class: net.bytebuddy.build.Plugin$Engine$Default
Export-Package: net.bytebuddy;version="1.9.3",net.bytebuddy.agent.builder;version="1.9.3",net.bytebuddy.asm;version="1.9.3",net.bytebuddy.build;version="1.9.3",net.bytebuddy.description;version="1.9.3",net.bytebuddy.description.annotation;version="1.9.3",net.bytebuddy.description.enumeration;version="1.9.3",net.bytebuddy.description.field;version="1.9.3",net.bytebuddy.description.modifier;version="1.9.3",net.bytebuddy.description.type;version="1.9.3",net.bytebuddy.dynamic;version="1.9.3",net.bytebuddy.dynamic.loading;version="1.9.3",net.bytebuddy.dyna...
1.525 proj4j 0.1.0

1.525.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

Copyright 2011 Martin Davis

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0.1-0-sources-jar/org/osgeo/proj4j/proj/Eckert6Projection.java
No license file was found, but licenses were detected in source scan.

/*

Copyright 2006 Jerry Huxtable

*/
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/PerspectiveConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/UrmaevFlatPolarSinusoidalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/TransverseCylindricalEqualArea.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/MercatorProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/HammerProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/BipolarProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/ConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/PlateCarreeProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/Wagner7Projection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/StereographicAzimuthalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/NicolsisProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/GallProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/MillerProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/WinkelTripelProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/AzimuthalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/CrasterProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/FoucautSinusoidalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/
jar/org/osgeo/proj4j/proj/GnomonicAzimuthalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/Eckert5Projection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/SineTangentSeriesProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/ObliqueStereographicAlternativeProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/McBrydeThomasFlatPolarQuarticProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/LandsatProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/EulerProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/EqualAreaAzimuthalProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/KavraiskyVProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/LinearProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/CollignonProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/DenoyerProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/LambertConformalConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/VanDerGrintenProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/CylindricalEqualAreaProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/PutninsP2Projection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/Murdoch2Projection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/EquidistantConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/EquidistantConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/ProjectionException.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/SimpleConicProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/AitoffProjection.java
* /opt/cola/permits/1202446574_1631568180.3/0/proj4j-0-1-0-sources-jar/org/osgeo/proj4j/proj/Wagner1Projection.java
Copyright 2006 Martin Davis

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright 2006 Jerry Huxtable

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
NNSS is available under the Mozilla Public License, version 2, a copy of which is below.

Note on GPL Compatibility
-----------------------------

The MPL 2, section 3.3, permits you to combine NSS with code under the GNU General Public License (GPL) version 2, or any later version of that license, to make a Larger Work, and distribute the result under the GPL. The only condition is that you must also make NSS, and any changes you have made to it, available to recipients under the terms of the MPL 2 also.

Anyone who receives the combined code from you does not have to continue to dual licence in this way, and may, if they wish, distribute under the terms of either of the two licences - either the MPL alone or the GPL alone. However, we discourage people from distributing copies of NSS under the GPL alone, because it means that any improvements they make cannot be reincorporated into the main version of NSS. There is never a need to do this for license compatibility reasons.

Note on LGPL Compatibility
-----------------------------

The above also applies to combining MPLed code in a single library with code under the GNU Lesser General Public License (LGPL) version 2.1, or any later version of that license. If the LGPLed code and the MPLed code are not in the same library, then the copyleft coverage of the two licences does not overlap, so no issues arise.
1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
   in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
   version 1.1 or earlier of the License, but not also under the
terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
Open Source Used In DNAC Platform Arica 2004

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions
---------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its
2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other
equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).
3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License.
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.
*                                                                      *
************************************************************************
************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This
limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous
-------------

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License
---------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions
If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------
This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Within this directory, each of the file listed below is licensed under
the terms given in the file LICENSE-MPL, also in this directory.

pi1k.txt
pi2k.txt
pi5k.txt
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.
Within this directory, each of the file listed below is licensed under
the terms given in the file LICENSE-MPL, also in this directory.

PRIMES
Within this directory, each of the file listed below is licensed under
the terms given in the file LICENSE-MPL, also in this directory.

basecvt.pod
gcd.pod
invmod.pod
isprime.pod
lap.pod
mpi-test.pod
prime.txt
prng.pod
# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Testing Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
1.527 xml2js-preserve-spaces 0.0.1 0.0.1

1.527.1 Available under license:

Copyright 2010, 2011, 2012, 2013. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.528 flat-cache 1.3.4
1.528.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Roy Riojas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.529 y18n 4.0.0 4.0.0
1.529.1 Available under license:
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.530 snappy-java 1.1.2.6

1.531 estraverse 4.3.0

1.531.1 Available under license:
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.532 semver 5.5.0 5.5.0

1.532.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.533 commons-digester 2.1

1.533.1 Available under license:

Apache Commons Digester
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.534 @babel/helper-split-export-declaration
7.11.0
1.534.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
1.535 libevent 1.3.6

1.535.1 Available under license:

CMake - Cross Platform Makefile Generator
Copyright 2000-2013 Kitware, Inc.
Copyright 2000-2011 Insight Software Consortium
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the names of Kitware, Inc., the Insight Software Consortium,
nor the names of their contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The above copyright and license notice applies to distributions of
CMake in source and binary form. Some source files contain additional
notices of original copyright by their contributors; see each source
Third-party software packages supplied with CMake under compatible licenses provide their own copyright notices documented in corresponding subdirectories.

------------------------------------------------------------------------------

CMake was initially developed by Kitware with the following sponsorship:

* National Library of Medicine at the National Institutes of Health as part of the Insight Segmentation and Registration Toolkit (ITK).

* US National Labs (Los Alamos, Livermore, Sandia) ASC Parallel Visualization Initiative.

* National Alliance for Medical Image Computing (NAMIC) is funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149.

* Kitware, Inc.

Libevent is available for use under the following license, commonly known as the 3-clause (or "modified") BSD license:

===============================================================================
Copyright (c) 2000-2007 Niels Provos <provos@citi.umich.edu>
Copyright (c) 2007-2012 Niels Provos and Nick Mathewson

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

Open Source Used In DNAC Platform Arica 2021
Portions of Libevent are based on works by others, also made available by them under the three-clause BSD license above. The copyright notices are available in the corresponding source files; the license is as above. Here's a list:

log.c:
   Copyright (c) 2000 Dug Song <dugsong@monkey.org>
   Copyright (c) 1993 The Regents of the University of California.

strlcpy.c:
   Copyright (c) 1998 Todd C. Miller <Todd.Miller@courtesan.com>

win32select.c:
   Copyright (c) 2003 Michael A. Davis <mike@datanerds.net>

evport.c:
   Copyright (c) 2007 Sun Microsystems

ht-internal.h:
   Copyright (c) 2002 Christopher Clark

minheap-internal.h:
   Copyright (c) 2006 Maxim Yegorushkin <maxim.yegorushkin@gmail.com>

The arc4module is available under the following, sometimes called the "OpenBSD" license:

Copyright (c) 1996, David Mazieres <dm@uun.org>
Copyright (c) 2008, Damien Miller <djm@openbsd.org>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED “AS IS” AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The Windows timer code is based on code from libutp, which is
Copyright (c) 2010 BitTorrent, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.536 object-assign 4.1.1 4.1.1

1.536.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.537 compress-lzf 1.1

1.537.1 Available under license:

# Compress LZF

This library contains efficient implementation of LZF compression format, as well as additional helper classes that build on JDK-provided gzip (deflat) codec.

## Licensing

Library is licensed under Apache License 2.0, as per accompanying LICENSE file.

## Credit

Library has been written by Tatu Saloranta (tatu.saloranta@iki.fi).
It was started at Ning, inc., as an official Open Source process used by platform backend, but after initial versions has been developed outside of Ning by supporting community.

Other contributors include:
* Jon Hartlaub (first versions of streaming reader/writer; unit tests)
* Cedrik Lime: parallel LZF implementation

Various community members have contributed bug reports, and suggested minor fixes; these can be found from file "VERSION.txt" in SCM.

This copy of Compress-LZF library is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.538 drill-udfs 1.19.0

1.538.1 Available under license:

Drill : Contrib : UDFs
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.539 is-negative-zero 2.0.1

1.539.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.540 regenerator-transform 0.10.1 0.10.1
1.540.1 Available under license :

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 *
 * This source code is licensed under the BSD-style license found in the
 * https://raw.githubusercontent.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */

MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.541 @babel/plugin-proposal-optional-catch-binding 7.16.0

1.541.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.542 @babel/plugin-transform-modules-amd 7.16.0

1.542.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.543 path-exists 4.0.0 4.0.0
1.543.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.544 lru-cache 6.0.0
1.544.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
1.545 grpc-core 1.27.1

1.545.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the
Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.546 cookie-parser 1.4.3 1.4.3

1.546.1 Available under license:

(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.547 swagger-annotations 1.6.2
1.547.1 Available under license :
No license file was found, but licenses were detected in source scan.

Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.html

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1088518024_1599800722.99/0/swagger-annotations-1-6-2-jar/META-INF/MANIFEST.MF

1.548 path-key 3.1.1
1.548.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.549 rimraf 3.0.2 3.0.2
1.549.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.550 babel-core 6.26.3 6.26.3

1.550.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.551 mime-types 2.1.30 2.1.30

1.551.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.552 progress 2.0.3 2.0.3

1.552.1 Available under license:
(The MIT License)

Copyright (c) 2017 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.553 okhttp 2.7.5

1.553.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.554 is-ci 2.0.0

1.554.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016-2018 Thomas Watson Steen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.555 glob-parent 2.0.0
1.555.1 Available under license :
   The ISC License

Copyright (c) 2015 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.556 ansi-styles 2.2.1
1.556.1 Available under license :
   The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.557 axios 0.18.1 0.18.1
1.557.1 Available under license:
Copyright (c) 2014-present Matt Zabriskie

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.558 activation-api 1.1
1.558.1 Available under license:
# Notices for Eclipse Project for JAF

This content is produced and maintained by the Eclipse Project for JAF project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaf

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0, which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaf

## Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.559 jnr-ffi 2.1.7
1.559.1 Available under license :
/*
 * Copyright (C) 2010 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.

Apache License
Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.560 hadoop-aws 3.2.2

1.560.1 Available under license:

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the
following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of
Gson 2.2.4,
which has the following notices:

The Netty Project
===================

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the
components that this product depends on.

------------------------------------------------------------------------
This product contains the extensions to Java Collections Framework which has
been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain
Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based
WebSocket and HTTP server, which can be obtained at:
This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
  * HOMEPAGE:
    * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
  * HOMEPAGE:
    * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
  * HOMEPAGE:
    * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of
a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression
and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/
This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of
Commons Codec 1.4,
which has the following notices:
* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from
  http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)
===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of
Commons Lang 2.6,
which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see:
  StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of
Apache Log4j 1.2.17,
which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative
  Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work
distributed in source code form must include this copyright and license
  notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
Jetty Web Container
Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
 ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api

Open Source Used In DNAC Platform Arica 2057
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL/GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/ (Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUntime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement
The binary distribution of this product bundles binaries of
Xerces2 Java Parser 2.9.1,
which has the following notices:

* =========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--

This product incorporates portions of the 'Hadoop' project
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.
Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.
http://glyphicons.com/

Copyright Jan Kovak
Licensed under the Apache License v2.0 as a part of the Bootstrap project.

This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors
Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).
This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
*=============================================================================
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
=============================================================================

This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64,
LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and 
limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright 
notices and license terms. Your use of the source code for the these 
subcomponents is subject to the terms and conditions of the following 
licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
* Copyright (c) 2005, European Commission project OneLab under contract
* 034819 (http://www.one-lab.org)
* All rights reserved.
* Redistribution and use in source and binary forms, with or
* without modification, are permitted provided that the following conditions are met:
  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  * Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++
 *
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 *
 * Abstract: The main routine
 *
 */

For src/main/native/src/org/apache/hadoop/io/compress/lz4/{lz4.h,lz4.c,lz4hc.h,lz4hc.c},

/*
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

 * Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following disclaimer
   in the documentation and/or other materials provided with the
   distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

*/

For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

---------------------------------------------------------------------

Open Source Used In DNAC Platform Arica 2069
This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------

Copyright 2002 Niels Provos <provos@citi.umich.edu>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.
This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=============

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
Open Source Used In DNAC Platform Arica 2074

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
--------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js
--------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js
--------------------------------------------------------------------------------
Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
- bootstrap v3.3.6
- broccoli-asset-rev v2.4.2
- broccoli-funnel v1.0.1
- datatables v1.10.8
- em-helpers v0.5.13
- em-table v0.1.6
- ember v2.2.0
- ember-array-contains-helper v1.0.2
- ember-bootstrap v0.5.1
- ember-cli v1.13.13
- ember-cli-app-version v1.0.0
- ember-cli-babel v5.1.6
- ember-cli-content-security-policy v0.4.0
- ember-cli-dependency-checker v1.2.0
- ember-cli-htmlbars v1.0.2
- ember-cli-htmlbars-inline-precompile v0.3.1
- ember-cli-ic-ajax v0.2.1
- ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCoding 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
delivered to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
/hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

--------------------------------------------------------------------------------
Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js

--------------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery
Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====
All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQLDB Database 2.3.4

(HSQL License)
"COPYRIGHTS AND LICENSES (based on BSD License)
For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its
contributors may be used to endorse or promote products derived from this
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the
Hypersonic SQL Group.”

The binary distribution of this product bundles these dependencies under the
following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1.Contributor means each individual or entity
that creates or contributes to the creation of
Modifications.

1.2.Contributor Version means the combination of the
Original Software, prior Modifications used by a Contributor (if any), and the
Modifications made by that particular Contributor.

1.3.Covered Software means (a) the Original Software, or (b) Modifications, or (c) the
combination of files containing Original Software with files containing
Modifications, in each case including portions thereof.

1.4.Executable means the Covered Software in any form other
than Source Code.

1.5.Initial Developer means the individual or entity
that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b),
above, no patent license is granted: (1) for any code that Contributor has
deleted from the Contributor Version; (2) for infringements caused by:
(i) third party modifications of Contributor Version, or (ii) the combination
of Modifications made by that Contributor with other software (except as part of
the Contributor Version) or other devices; or (3) under Patent Claims infringed
by Covered Software in the absence of Modifications made by that
Contributor.

3. Distribution Obligations.

3.1. Availability of Source
Code. Any Covered Software that You distribute or otherwise make available in
Executable form must also be made available in Source Code form and that Source
Code form must be distributed only under the terms of this License. You must
include a copy of this License with every copy of the Source Code form of the
Covered Software You distribute or otherwise make available. You must inform
recipients of any such Covered Software in Executable form as to how they can
obtain such Covered Software in Source Code form in a reasonable manner on or
through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are
governed by the terms of this License. You represent that You believe Your
Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices. You must
include a notice in each of Your Modifications that identifies You as the
Contributor of the Modification. You may not remove or alter any copyright,
patent or trademark notices contained within the Covered Software, or any
notices of licensing or any descriptive text giving attribution to any
Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code
form that alters or restricts the applicable version of this License or the
recipients rights hereunder. You may choose to offer, and to charge a fee for,
warranty, support, indemnity or liability obligations to one or more recipients
of Covered Software. However, you may do so only on Your own behalf, and not on
behalf of the Initial Developer or any Contributor. You must make it absolutely
clear that any such warranty, support, indemnity or liability obligation is
offered by You alone, and You hereby agree to indemnify the Initial Developer
and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of warranty, support, indemnity or liability terms
You offer.

3.5. Distribution of Executable Versions. You may distribute the
Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER
COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby
grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution...
to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software.
Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or
indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERICAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding
such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04
--------------------------------------------------------------------------------

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1
--------------------------------------------------------------------------------

(EPL v1.0)
Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2
--------------------------------------------------------------------------------
(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.
--------------------------------------------------------------------------------
(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this
License.
"Licensor" means the individual or entity that offers the Work under the terms of this License.
"Original Author" means the individual or entity who created the Work.
"Work" means the copyrightable work of authorship offered under the terms of this License.
"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications
All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING
THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not
be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

------------------------------------------------------------------------------------------------------------------------

(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributors Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.
1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the
time of the initial grant or subsequently, any and all of the rights conveyed by
this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or
modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and
apparatus claims, in any patent Licensable by such Contributor that would be
infringed, but for the grant of the License, by the making, using, selling,
offering for sale, having made, import, or transfer of either its Contributions
or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General
Public License, Version 2.1, the GNU Affero General Public License, Version 3.0,
or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For
legal entities, You includes any entity that controls, is controlled by,
or is under common control with You. For purposes of this definition,
control means (a) the power, direct or indirect, to cause the direction or
management of such entity, whether by contract or otherwise, or (b) ownership of
more than fifty percent (50%) of the outstanding shares or beneficial ownership
of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:

under intellectual property rights (other than patent or trademark) Licensable
by such Contributor to use, reproduce, make available, modify, display, perform,
distribute, and otherwise exploit its Contributions, either on an unmodified
basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have
made, import, and otherwise transfer either its Contributions or its Contributor
Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become
effective for each Contribution on the date the Contributor first distributes
such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this
License. No additional rights or licenses will be implied from the distribution
or licensing of Covered Software under this License. Notwithstanding Section
2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's
modifications of Covered Software, or (ii) the combination of its Contributions
with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its
Contributions.

This License does not grant any rights in the trademarks, service marks, or
logos of any Contributor (except as may be necessary to comply with the notice
requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute
the Covered Software under a subsequent version of this License (see Section
10.2) or under the terms of a Secondary License (if permitted under the terms of
Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are
its original creation(s) or it has sufficient rights to grant the rights to its
Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable
copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including
copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.
5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be
construed against the drafter shall not be used to construe this License against
a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3,
no one other than the license steward has the right to modify or publish new
versions of this License. Each version will be given a distinguishing version
number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the
License under which You originally received the Covered Software, or under the
terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a
new license for such software, you may create and use a modified version of this
License if you rename the license and remove any references to the name of the
license steward (except to note that such modified license differs from this
License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary
Licenses under the terms of this version of the License, the notice described in
Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v.
2.0. If a copy of the MPL was not distributed with this file, You can obtain one
at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then
You may include the notice in a location (such as a LICENSE file in a relevant
directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined
by the Mozilla Public License, v. 2.0.
The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

--------------------------------------------------------------------------------
/*--
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes
cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a
performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:
i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.
b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g., Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer
UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and
enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.561 parse-glob 3.0.4

1.561.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.562 supports-color 5.5.0 5.5.0

1.562.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.563 form-data 2.5.0 2.5.0

1.563.1 Available under license:
Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.564 yauaa 5.20

1.564.1 Available under license:

No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an AS IS BASIS,

Found in path(s):
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/LG.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Opera.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GoogleChrome.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GameConsoles.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Bada.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Alcatel.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Watches.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/UserAgents/Robots-PriceSpiders.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GoogleFavicon.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Windows.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/MobileBrand-rules.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Tencent.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Lenovo-Tests.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Huawei.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Ubuntu-tests.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Rots.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Robots-Google.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/OtherOS.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Linux.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Cubot.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Nokia.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GoogleChromeFrozen.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/WindowsCE.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/MSEdge.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GoogleChrome70.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Apple.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Hackers.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/WindowsPhone.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Amazon.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GeckoFirefox.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/TV.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/GoannaPaleMoon.yaml`
* `/opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/UUID.yaml`
No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/OperatingSystemTests.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/AppleTypes.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/WindowsOEMCodes.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/WindowsPhoneLookups.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/UbuntuVersionLookups.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/OperatingSystemDeviceNames.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/MobileBrands.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Ubuntu.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/Ubuntu-18.10-tests.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/CPUTypes.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/AmazonDevices.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/RobotBaseRules.yaml
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/UserAgents/AndroidDeviceOverrides.yaml

No license file was found, but licenses were detected in source scan.

--

2020 Niels Basjes

~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ https://www.apache.org/licenses/LICENSE-2.0

~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.

Found in path(s):
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/META-INF/maven/nl.basjes.parse.useragent/yauaa/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Yet Another UserAgent Analyzer
* Copyright (C) 2013-2020 Niels Basjes
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

Open Source Used In DNAC Platform Arica 2125
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/debug/FlattenPrinter.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/UserAgent.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/annotate/UserAgentAnnotationMapper.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/walk/StepUp.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/walk/StepDown.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/compare/StepEndsWith.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/compare/StepDefaultIfNull.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/compare/StepIsNull.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/classify/DeviceClass.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/CalculateAgentEmail.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/ConcatNONDuplicatedCalculator.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/CalculateDeviceBrand.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/utils/YauaaVersion.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/lookup/StepLookupPrefix.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/compare/StepDefaultIfNull.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/annotate/UserAgentAnnotationMapper.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/walk/StepUp.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/walk/StepDown.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/ConcatNONDuplicatedCalculator.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/analyze/treewalker/steps/compare/StepDefaultIfNull.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/CalculateDeviceBrand.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-jar/nl/basjes/parse/useragent/calculate/CalculateAgentEmail.java
* /opt/cola/permits/1202446250_1631568086.76/0/yauaa-5-20-sources-
1.565 batik-awt-util 1.13

1.565.1 Available under license:

Apache Batik

Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the

Open Source Used In DNAC Platform Arica  2129
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for Standardization for the definition of character entities used in the software's documentation.

This product includes images from the Tango Desktop Project (http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme (http://www.jesusda.com/projects/pasodoble).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.566 pi-spec-converter-web 0.0.0 0.0.0

1.566.1 Available under license:

  core-js@2.5.1
  MIT
  Copyright (c) 2014-2017 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

webpack@3.6.0
MIT
Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

zone.js@0.8.18
MIT
The MIT License

Copyright (c) 2016 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

@angular/core@4.4.5
MIT
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

webpack@3.6.0
MIT
Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

zone.js@0.8.18
MIT
The MIT License

Copyright (c) 2016 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

@angular/core@4.4.5
MIT
MIT

@angular/http@4.4.5
MIT
MIT

file-saver@1.3.3
MIT
The MIT License

Copyright 2016 [Eli Grey][1].

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[1]: http://eligrey.com

@angular/forms@4.4.5
MIT
MIT

@angular/platform-browser@4.4.5
MIT
MIT

@angular/common@4.4.5
MIT
MIT

(MIT License)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.567 @babel/runtime 7.16.3

1.567.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.568 async-each 1.0.3

1.568.1 Available under license:

No license file was found, but licenses were detected in source scan.

// async-each MIT license (by Paul Miller from https://paulmillr.com).
The MIT License (MIT)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal to use, copy, modify, merge, publish, distribute, sublicense, and/or sell furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

1.569 moment-timezone 0.5.33 0.5.33
1.569.1 Available under license :

The MIT License (MIT)

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.570 amdefine 1.0.1
1.570.1 Available under license:

Dojo is available under *either* the terms of the modified BSD license *or* the Academic Free License version 2.1. As a recipient of Dojo, you may choose which license to receive this code under (except as noted in per-module LICENSE files). Some modules may not be the copyright of the Dojo Foundation. These modules contain explicit declarations of copyright in both the LICENSE files in the directories in which they reside and in the code itself. No external contributions are allowed under licenses which are fundamentally incompatible with the AFL or BSD licenses that Dojo is distributed under.

The text of the AFL and BSD licenses is reproduced below.

-------------------------------------------------------------------------------
The "New" BSD License:
**************************

Copyright (c) 2005-2009, The Dojo Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-------------------------------------------------------------------------------

The Academic Free License, v. 2.1:
*************************************************

This Academic Free License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the
Licensed under the Academic Free License version 2.1

1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:

a) to reproduce the Original Work in copies;

b) to prepare derivative works ("Derivative Works") based upon the Original Work;

c) to distribute copies of the Original Work and Derivative Works to the public;

d) to perform the Original Work publicly; and

e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license, under patent claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.

4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if
such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to license.

5) This section intentionally omitted.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries, and by international treaty. Therefore, by
exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, against Licensor or any licensee alleging that the Original Work infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware.

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. 101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys' fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.

13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.
This license is Copyright (C) 2003-2004 Lawrence E. Rosen. All rights reserved. Permission is hereby granted to copy and distribute this license without modification. This license may not be modified without the express written permission of its copyright owner.

amdefine is released under two licenses: new BSD, and MIT. You may pick the license that best suits your development needs. The text of both licenses are provided below.

The "New" BSD License:
----------------------

Copyright (c) 2011-2016, The Dojo Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License
-----------

Copyright (c) 2011-2016, The Dojo Foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.571 unicode-match-property-ecmascript
2.0.0
1.571.1 Available under license :
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.572 import-lazy 2.1.0
1.572.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.573 is-buffer 1.1.6

1.573.1 Available under license:

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
### 1.574 package-hash 4.0.0

#### 1.574.1 Available under license:

**ISC License (ISC)**

Copyright (c) 2016-2017, Mark Wubben <mark@novemberborn.net> (novemberborn.net)

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

### 1.575 v8flags 3.2.0

#### 1.575.1 Available under license:

**The MIT License (MIT)**

Copyright (c) 2014-2018 Tyler Kellen <tyler@sleekcode.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.576 string-width 4.2.3

1.576.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.577 socket.io-adapter 1.1.2 1.1.2

1.577.1 Available under license:

(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.578 ajv-keywords 1.5.1

1.578.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.579 get-func-name 2.0.0

1.579.1 Available under license:

Copyright (c) 2013 Jake Luer <jake@alogicalparadox.com> (http://alogicalparadox.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.580 jsreport-express 2.8.1 2.8.1
1.580.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.
1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities,
conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.581 httpasyncclient 4.1.4

1.581.1 Available under license:

Apache HttpAsyncClient
Copyright 2010-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
“License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.582 hogan 3.0.2 3.0.2

1.582.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.583 hikaricp 3.4.2
1.583.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Brett Wooldridge
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*

Copyright (C) 2013, 2014 Brett Wooldridge

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
you may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):

/*

Copyright (C) 2013, 2014 Brett Wooldridge

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
you may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):

/*

Copyright (C) 2013, 2014 Brett Wooldridge

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
you may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/
jar/com/zaxxer/hikari/util/ConcurrentBag.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/pool/ProxyLeakTask.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/pool/ProxyLeakTaskFactory.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/prometheus/HikariCPCollector.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/util/SuspendResumeLock.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/HikariConfig.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2016 Brett Wooldridge
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/prometheus/PrometheusMetricsTrackerFactory.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2013,2014 Brett Wooldridge
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/pool/HikariPool.java
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/metrics/dropwizard/CodaHaleMetricsTracker.java
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/metrics/MetricsTracker.java
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/metrics/MetricsTrackerFactory.java
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/HikariJNDIFactory.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013, 2014 Brett Wooldridge
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/com/zaxxer/hikari/pool/PoolBase.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Brett Wooldridge
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/util/IsolationLevel.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 Brett Wooldridge
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/pool/ProxyStatement.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/dropwizard/CodahaleMetricsTrackerFactory.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/pool/ProxyPreparedStatement.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/HikariDataSource.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/HikariPoolMXBean.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/prometheus/PrometheusHistogramMetricsTracker.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/hibernate/HikariConnectionProvider.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/hibernate/HikariConfigurationUtil.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/util/UtilityElf.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/util/PropertyElf.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-
jar/com/zaxxer/hikari/HikariConfigMXBean.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Brett Wooldridge
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/IMetricsTracker.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Brett Wooldridge
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/metrics/dropwizard/CodahaleHealthChecker.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-jar/com/zaxxer/hikari/pool/PoolEntry.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 Brett Wooldridge
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-
  jar/com/zaxxer/hikari/util/ClockSource.java
* /opt/cola/permits/1257346932_1642803438.49/0/hikaricp-3-4-2-sources-
  jar/com/zaxxer/hikari/metrics/PoolStats.java

1.584 chokidar 3.5.2
1.584.1 Available under license:
The MIT License (MIT)

Copyright (c) 2012-2019 Paul Miller (https://paulmillr.com), Elan Shanker

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the Software), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.585 logback-core 1.2.3

1.585.1 Available under license :

Found license 'Eclipse Public License 1.0' in "* Copyright (C) 1999-2015, QOS.ch. All rights reserved. * This program and the accompanying materials are dual-licensed under * either the terms of the Eclipse Public License v1.0 as published by * under the terms of the GNU Lesser General Public License version 2.1 * as published by the Free Software Foundation.'

Found license 'GNU Lesser General Public License' in "* Copyright (C) 1999-2015, QOS.ch. All rights reserved. * This program and the accompanying materials are dual-licensed under * either the terms of the Eclipse Public License v1.0 as published by * under the terms of the GNU Lesser General Public License version 2.1 * as published by the Free Software Foundation.'

1.586 mime 1.3.4 1.3.4

1.586.1 Available under license :

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.587 jline-builtins 3.12.1

1.587.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2016, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/o/jline-builtsins-3-12-1-sources-jar/org/jline/builtsins/TTop.java
* /opt/cola/permits/1114600151_1607533999.81/o/jline-builtsins-3-12-1-sources-jar/org/jline/builtsins/Tmux.java
* /opt/cola/permits/1114600151_1607533999.81/o/jline-builtsins-3-12-1-sources-jar/org/jline/builtsins/Source.java
* /opt/cola/permits/1114600151_1607533999.81/o/jline-builtsins-3-12-1-sources-jar/org/jline/builtsins/Commands.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2016, the original author or authors.
 * This software is distributable under the BSD license. See the terms of the 
 * BSD license in the documentation provided with this software.
 */

https://opensource.org/licenses/BSD-3-Clause

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more 
 * contributor license agreements. See the NOTICE file distributed with 
 * this work for additional information regarding copyright ownership. 
 * The ASF licenses this file to You under the Apache License, Version 2.0 
 * (the "License"); you may not use this file except in compliance with 
 * the License. You may obtain a copy of the License at 
 */

http://www.apache.org/licenses/LICENSE-2.0

/* Unless required by applicable law or agreed to in writing, software 
 * distributed under the License is distributed on an "AS IS" BASIS, 
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
 * See the License for the specific language governing permissions and 
 * limitations under the License. */

/*
 * Based on http://antony.lesuisse.org/software/ajaxterm/
 * Public Domain License
 */

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/o/jline-builtsins-3-12-1-sources-jar/org/jline/builtsins/ScreenTerminal.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2018, the original author or authors.
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
*
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/NfaMatcher.java
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/Completers.java
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/InputRC.java
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/Less.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2017, the original author or authors.
*
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
*
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/Nano.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2018, the original author or authors.
*
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
*
* https://opensource.org/licenses/BSD-3-Clause
*/

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*/
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/org/jline/builtins/Options.java
No license file was found, but licenses were detected in source scan.

2016, the original author or authors.

This software is distributable under the BSD license. See the terms of the
BSD license in the documentation provided with this software.

https://opensource.org/licenses/BSD

Found in path(s):
* /opt/cola/permits/1114600151_1607533999.81/0/jline-builtins-3-12-1-sources-jar/META-INF/maven/org.jline/jline-builtins/pom.xml

1.588 orc-core 1.5.1
1.588.1 Available under license:

ORC Core
Copyright 2013-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.589 failureaccess 1.0.1
1.589.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1199742917_1630577017.47/0/failureaccess-1-0-1-sources-3-
jar/com/google/common/util/concurrent/internal/InternalFutureFailureAccess.java
* /opt/cola/permits/1199742917_1630577017.47/0/failureaccess-1-0-1-sources-3-
jar/com/google/common/util/concurrent/internal/InternalFutures.java

1.590 commons-crypto 1.0.0

1.591 htrace-core4 4.1.0-incubating
1.591.1 Available under license:

htrace-core4
Copyright 2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.592 debug 4.3.1

1.592.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.593 is 3.3.0 3.3.0

1.593.1 Available under license:
(The MIT License)

Copyright (c) 2013 Enrico Marino
Copyright (c) 2014 Enrico Marino and Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.594 istanbul-lib-report 3.0.0
1.594.1 Available under license:
MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.595 json-schema-ref-parser 3.3.1 3.3.1
1.595.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 James Messinger

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.596 babel-plugin-transform-es2015-shorthand-properties 6.24.1 6.24.1

1.596.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.597 re2j 1.1

1.598 aws4 1.10.1 1.10.1
Available under license:

Copyright 2013 Michael Hart (michael.hart.au@gmail.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

clone-deep 4.0.1

Available under license:

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.600 chai-http 3.0.0
1.600.1 Available under license :
MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE

1.601 lcms 2.6 3ubuntu2.1
1.601.1 Available under license :
Little CMS
Copyright (c) 1998-2011 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THIS LICENSE APPLIES ONLY TO iccjpeg.c file
-----

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-2013, Thomas G. Lane, Guido Vollbeding. All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:
(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".
(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.
The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, lttmain.sh). Another support script, install-sh, is copyright by X Consortium but is also freely distributable.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that
"The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

1.602 stack-trace 0.0.10 0.0.10
1.602.1 Available under license:
Copyright (c) 2011 Felix Geisendrfer (felix@debuggable.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.603 jcommander 1.30
1.603.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2011 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
jar/com/beust/jcommander/IParameterValidator.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
jar/com/beust/jcommander/validators/PositiveInteger.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
jar/com/beust/jcommander/validators/IParameterValidator2.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
jar/com/beust/jcommander/validators/NoValidator.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
jar/com/beust/jcommander/validators/NoValueValidator.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2010 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/IntegerConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/ParametersDelegate.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/MissingCommandException.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/ResourceBundle.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/BooleanConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/FloatConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/internal/DefaultConverterFactory.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/JCommander.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/NoConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/Parameter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/defaultprovider/PropertyFileDefaultProvider.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/LongConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/BaseConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/IStringConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/IStringConverterFactory.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/ISO8601DateConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/ParameterException.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/IDefaultProvider.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/StringConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
  jar/com/beust/jcommander/converters/DoubleConverter.java
* /opt/cola/permits/1136701934_1613865330.83/0/jcommander-1-30-sources-1-
1.604 content-disposition 0.5.2 0.5.2
1.604.1 Available under license:
(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.605 stealthy-require 1.1.1 1.1.1
1.605.1 Available under license:
ISC License

Copyright (c) 2017, Nicolai Kamenzky and contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.606 node-preload 0.2.1

1.606.1 Available under license:
MIT License

Copyright (c) 2019 CFWare, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.607 sqlite 3.21.0

1.607.1 Available under license:
The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.
1.608 lodash.clonedep 4.5.0 4.5.0

1.608.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.609 is-extendable 1.0.1 1.0.1

1.609.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.610 commons-compress 1.19

1.610.1 Available under license:

Apache Commons Compress

Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (https://www.apache.org/).

The files in the package org.apache.commons.compress.archivers.sevenz were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/), which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
1.611 xmlbuilder 11.0.1

1.611.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013 Ozgur Ozcitak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.612 drill-rpc 1.19.0

1.612.1 Available under license:

Drill : Exec : RPC

Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner...
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.613 httpcore-nio 4.4.6
1.613.1 Available under license:

Apache HttpCore NIO
Copyright 2005-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.614 extend-shallow 3.0.2 3.0.2

1.614.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1.615 accepts 1.3.5 1.3.5
1.615.1 Available under license :
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.616 punycode 2.1.1 2.1.1
1.616.1 Available under license :
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.617 my-scope-package-b 0.0.0

1.617.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The JsonCpp library's source code, including accompanying documentation, tests and demonstration applications, are licensed under the following conditions...

The author (Baptiste Lepilleur) explicitly disclaims copyright in all
jurisdictions which recognize such a disclaimer. In such jurisdictions, this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of 2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur, and is released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this software may choose to accept it either as 1) Public Domain, 2) under the conditions of the MIT License (see below), or 3) under the terms of dual Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

========================================================================
Copyright (c) 2007-2010 Baptiste Lepilleur

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
========================================================================
(END LICENSE TEXT)

The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the
Public Domain "license" you can re-license your copy using whatever license you like.
All files in this directory are provided by the following license if not stated otherwise in the individual file:

This file is provided under a dual BSD/GPLv2 license. When using or redistributing this file, you may do so under either license.

GPL LICENSE SUMMARY

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.

This program is free software; you can redistribute it and/or modify it under the terms of version 2 of the GNU General Public License as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St - Fifth Floor, Boston, MA 02110-1301 USA.
The full GNU General Public License is included in this distribution in the file called LICENSE.GPL.

Contact Information:

BSD LICENSE

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This license applies to all parts of V8 that are not externally maintained libraries. The externally maintained libraries used by V8 are:

- PCRE test suite, located in
  test/mjsunit/third_party/regexp-pcre/regexp-pcre.js. This is based on the test suite from PCRE-7.3, which is copyrighted by the University of Cambridge and Google, Inc. The copyright notice and license are embedded in regexp-pcre.js.

- Layout tests, located in test/mjsunit/third_party/object-keys. These are based on layout tests from webkit.org which are copyrighted by Apple Computer, Inc. and released under a 3-clause BSD license.

- Strongtalk assembler, the basis of the files assembler-arm-inl.h, assembler-arm.cc, assembler-arm.h, assembler-ia32-inl.h, assembler-ia32.cc, assembler-ia32.h, assembler-x64-inl.h, assembler-x64.cc, assembler-x64.h, assembler-mips-inl.h, assembler-mips.cc, assembler-mips.h, assembler.cc and assembler.h. This code is copyrighted by Sun Microsystems Inc. and released under a 3-clause BSD license.

- Valgrind client API header, located at src/third_party/valgrind/valgrind.h
  This is released under the BSD license.

- The Wasm C/C++ API headers, located at third_party/wasm-api/wasm.{h,hh}
  This is released under the Apache license. The API's upstream prototype implementation also formed the basis of V8's implementation in src/wasm/c-api.cc.

These libraries have their own licenses; we recommend you read them, as their terms may differ from the terms below.

Further license information can be found in LICENSE files located in sub-directories.
Copyright 2014, the V8 project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1994-2006 Sun Microsystems Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Sun Microsystems or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;

ii. moral rights retained by the original author(s) and/or performer(s);

iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;

v. rights protecting the extraction, dissemination, use and reuse of data in a Work;

vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and

vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining
Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

For more information, please see <http://creativecommons.org/publicdomain/zero/1.0/>
Tests included here are based on build output generated by the six-speed benchmark suite.

Copyright (c) 2015 Kevin Decker

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Notice that the following BSD-style license applies to this one file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

This file is part of Valgrind, a dynamic binary instrumentation framework.

Copyright (C) 2000-2010 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Notice that the above BSD-style license applies to this one file (valgrind.h) only. The entire rest of Valgrind is licensed under the terms of the GNU General Public License, version 2. See the COPYING file in the source distribution for details.

// Copyright 2016 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

PCRE LICENCE
------------
PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 7 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions.
THE BASIC LIBRARY FUNCTIONS

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2007 University of Cambridge
All rights reserved.

THE C++ WRAPPER FUNCTIONS

Contributed by: Google Inc.

Copyright (c) 2007, Google Inc.
All rights reserved.

THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End
Copyright (c) 2006 Apple Computer, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder(s) nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010 Jonathan Hartley
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors
  may be used to endorse or promote products derived from this software without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

A. HISTORY OF THE SOFTWARE
=====================================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations, which became
Zope Corporation. In 2001, the Python Software Foundation (PSF, see
https://www.python.org/psf/) was formed, a non-profit organization
created specifically to own Python-related Intellectual Property.
Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for
the Open Source Definition). Historically, most, but not all, Python
releases have also been GPL-compatible; the table below summarizes
the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td>1.2</td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
</tr>
<tr>
<td>1.3 thru 1.5.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>Version</td>
<td>Release Date</td>
<td>Company</td>
<td>License Type</td>
<td>Support</td>
</tr>
<tr>
<td>---------</td>
<td>--------------</td>
<td>---------</td>
<td>--------------</td>
<td>---------</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2 and above</td>
<td>2.1.1</td>
<td>2001-now</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.


3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGEMENT ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
-------------------------------------------------

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an
office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the
Individual or Organization ("Licensee") accessing and otherwise using
this software in source or binary form and its associated
documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive,
royalty-free, world-wide license to reproduce, analyze, test, perform
and/or display publicly, prepare derivative works, distribute, and
otherwise use the Software alone or in any derivative version,
provided, however, that the BeOpen Python License is retained in the
3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
---------------------------------------

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c)
1995-2001 Corporation for National Research Initiatives; All Rights Reserved” are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): “Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013”.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.
8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

--------------------------------------------------

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (C) 1993-2004 by Sun Microsystems, Inc. All rights reserved.

Developed at SunSoft, a Sun Microsystems, Inc. business.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

----------------------------------------------------------------

Notice that the following BSD-style license applies to this one file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

----------------------------------------------------------------

This file is part of Valgrind, a dynamic binary instrumentation framework.
Copyright (C) 2000-2007 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

W3C 3-clause BSD License

http://www.w3.org/Consortium/Legal/2008/03-bsd-license.html

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of works must retain the original copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the original copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the W3C nor the names of its contributors may be used to endorse or promote products derived from this work without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2006-2011, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.618 javax-el 2.2.4
1.618.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
Version 1.0

* 

1. Definitions.

  1.1. "Contributor? means each individual or entity that creates or contributes to the creation of Modifications.

  1.2. "Contributor Version? means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

  1.3. "Covered Software? means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.


  1.5. "Initial Developer? means the individual or entity that first makes Original Software available under this License.

  1.6. "Larger Work? means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

  1.7. "License? means this document.

  1.8. "Licensable? means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

  1.9. "Modifications? means the Source Code and Executable form of any of the following:

  A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

  B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. ?Original Software? means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. ?Patent Claims? means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. ?Source Code? means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. ?You? (or ?Your?) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, ?You? includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, ?control? means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You
represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients? rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient?s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a...
Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY?S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a ?commercial item,? as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of ?commercial computer software? (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and ?commercial computer software documentation? as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction?s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts.
located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

1.619 batik-rasterizer 1.13

1.619.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1146690630_1616599309.0/0/batik-rasterizer-1-13-sources-1-jar/META-INF/maven/org.apache.xmlgraphics/batik-rasterizer/pom.xml

No license file was found, but licenses were detected in source scan.

// -----------------------------------------------------------------------------
//
// Licensed to the Apache Software Foundation (ASF) under one or more
// contributor license agreements. See the NOTICE file distributed with
grant codeBase "${app.jar.base}/xml-apis-ext-${xmlapis.version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/xml-apis-${xmlapis.version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/xalan-${xalan.version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/xercesImpl-2.9.1.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/../batik-rasterizer-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-svgrasterizer-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-ext-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-dom-${version}.jar" {
    permission java.security.AllPermission;
};
grant codeBase "${app.jar.base}/batik-css-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-svg-dom-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-gvt-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-parser-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-script-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-bridge-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-swing-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-transcoder-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-gui-util-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-awt-util-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-util-${version}.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-xml-${version}.jar" {
    permission java.security.AllPermission;
};
grant codeBase ":${app.jar.base}/fop-transcoder-allinone-$\{xmlgraphics.fop.version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/xmlgraphics-commons-$\{xmlgraphics.commons.version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/batik-anim-$\{version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/batik-codec-$\{version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/batik-extension-$\{version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/batik-svggen-$\{version\}.jar" {
  permission java.security.AllPermission;
};

grant codeBase ":${app.jar.base}/js.jar" {
  permission java.lang.RuntimePermission "createClassLoader";
  permission java.net.SocketPermission "*", "listen, connect, resolve, accept";
  permission java.lang.RuntimePermission "accessDeclaredMembers";
  permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapter", "read";
  permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapterClassName", "read";
  permission java.io.FilePermission "<<ALL FILES>>", "read";
};

grant {
  permission java.io.FilePermission ":${app.jar.base}/batik-svg-dom-$\{version\}.jar", "read";
};

Found in path(s):
* /opt/cola/permits/1146690630_1616599309.0/0/batik-rasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/resources/rasterizer.policy

1.620 jnr-netdb 0.4
1.620.1 Available under license:

This code is free software: you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 3 only, as
published by the Free Software Foundation.

This code is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License
version 3 for more details.

You should have received a copy of the GNU Lesser General Public License
version 3 along with this work. If not, see <http://www.gnu.org/licenses/>.

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to
ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would otherwise
be required to provide such information under section 6 of the
GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the
Combined Work produced by recombining or relinking the
Application with a modified version of the Linked Version. (If
you use option 4d0, the Installation Information must accompany
the Minimal Corresponding Source and Corresponding Application
Code. If you use option 4d1, you must provide the Installation
Information in the manner specified by section 6 of the GNU GPL
for conveying Corresponding Source.)


You may place library facilities that are a work based on the
Library side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based
on the Library, uncombined with any other library facilities,
conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the
accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the
Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.
To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system.
(if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.
3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no
permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright
holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation
(including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a
covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.
14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year>  <name of author>  
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

### 1.621 lucene 7.4.0

#### 1.621.1 Available under license:

# JTS Topology Suite Licensing

May 30th, 2017

## Project Licenses

The Eclipse Foundation makes available all content in this project ("Content"). Unless otherwise indicated below, the Content is provided to you under the terms and conditions of either the [Eclipse Public License 1.0](https://www.eclipse.org/legal/epl-v10.html) ("EPL") or the [Eclipse Distribution License 1.0](http://www.eclipse.org/org/documents/edl-v10.php) (a BSD Style License). For purposes of the EPL, "Program" will mean the Content.

If you did not receive this Content directly from the Eclipse Foundation, the Content is being redistributed by another party ("Redistributor") and different terms and conditions may apply to your use of any object code in the Content. Check the Redistributor's license that was provided with the Content. If no such license exists, contact the Redistributor. Unless otherwise indicated below, the terms and conditions of the EPL still apply to any source code in the Content and such source code may be obtained at http://www.eclipse.org.

## Third Party Content

The Content includes items that have been sourced from third parties as set out below. If you did not receive this Content directly from the Eclipse Foundation, the following is provided for informational purposes only, and you should look to the Redistributor's license for terms and conditions of use.

### Jdom-contrib version 1.1.3

The JTS tests module contains source from JDOM Contrib 1.1.3 which is available under the Jdom License [1]. The
two files are LineNumberElement.java and LineNumberSAXHandler.java.

Typical users of this library will not need to depend on this module.


### GeoTools

JTS includes some code from the GeoTools project. This code has been licensed to the JTS project under the OSGeo BSD License [2] by the GeoTools PSC [3,4].


Eclipse.org Software User Agreement

March 17, 2005

Usage Of Content

THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW, THEN YOU MAY NOT USE THE CONTENT.

Applicable Licenses

Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is provided with this Content and is also available at http://www.eclipse.org/legal/epl-v10.html.

For purposes of the EPL, "Program" will mean the Content.
Content includes, but is not limited to, source code, object code, documentation and other files maintained in the Eclipse.org CVS repository ("Repository") in CVS modules ("Modules") and made available as downloadable archives ("Downloads").<p></p>

- Content may be structured and packaged into modules to facilitate delivering, extending, and upgrading the Content. Typical modules may include plug-ins ("Plug-ins"), plug-in fragments ("Fragments"), and features ("Features").</li>
- Each Plug-in or Fragment may be packaged as a sub-directory or JAR (Java ARchive) in a directory named "plugins".</li>
- Feature is a bundle of one or more Plug-ins and/or Fragments and associated material. Each Feature may be packaged as a sub-directory in a directory named "features". Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of the Plug-ins and/or Fragments associated with that Feature.</li>
- Features may also include other Features ("Included Features"). Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of Included Features.</li>

The terms and conditions governing Plug-ins and Fragments should be contained in files named "about.html" ("Abouts"). The terms and conditions governing Features and Included Features should be contained in files named "license.html" ("Feature Licenses"). Abouts and Feature Licenses may be located in any directory of a Download or Module including, but not limited to the following locations:</li>

- The top-level (root) directory
- Plug-in and Fragment directories
- Inside Plug-ins and Fragments packaged as JARs
- Sub-directories of the directory named "src" of certain Plug-ins
- Feature directories

Note: if a Feature made available by the Eclipse Foundation is installed using the Eclipse Update Manager, you must agree to a license ("Feature Update License") during the installation process. If the Feature contains Included Features, the Feature Update License should either provide you with the terms and conditions governing the Included Features or inform you where you can locate them. Feature Update Licenses may be found in the "license" property of files named "feature.properties"
Such Abouts, Feature Licenses, and Feature Update Licenses contain the terms and conditions (or references to such terms and conditions) that govern your use of the associated Content in that directory.<p>
</p>

THE ABOUTS, FEATURE LICENSES, AND FEATURE UPDATE LICENSES MAY REFER TO THE EPL OR OTHER LICENSE AGREEMENTS, NOTICES OR TERMS AND CONDITIONS. SOME OF THESE OTHER LICENSE AGREEMENTS MAY INCLUDE (BUT ARE NOT LIMITED TO):</ul>

<ul>
<li>Apache Software License 2.0 (available at <a href="http://www.apache.org/licenses/LICENSE-2.0">http://www.apache.org/licenses/LICENSE-2.0</a>)</li>
</ul>

IT IS YOUR OBLIGATION TO READ AND ACCEPT ALL SUCH TERMS AND CONDITIONS PRIOR TO USE OF THE CONTENT. If no About, Feature License, or Feature Update License is provided, please contact the Eclipse Foundation to determine what terms and conditions govern that particular Content.<p>
</p>

Cryptography<h3>
</h3>

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.<p>
</p>

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. <small>© 2003</small>
</body></html>
Copyright 1991-2018 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER Resulting FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.


# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
# # Redistributions of source code must retain the above copyright notice,
# # this list of conditions and the following disclaimer.
# # Redistributions in binary form must reproduce the above
# # copyright notice, this list of conditions and the following
# # disclaimer in the documentation and/or other materials provided with
# # the distribution.
Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The word list in cjdict.txt are generated by combining three word lists listed below with further processing for compound word breaking. The frequency is generated with an iterative training against Google web corpora.

* Libtabe (Chinese)
  - https://sourceforge.net/project/?group_id=1519
  - Its license terms and conditions are shown below.

* IPADIC (Japanese)
  - http://chasen.aist-nara.ac.jp/chasen/distribution.html
  - Its license terms and conditions are shown below.

---------COPYING.libtabe ---- BEGIN-------------------

/*
 * Copyright (c) 1999 TaBE Project.
 * Copyright (c) 1999 Pai-Hsiang Hsiao.
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * . Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the

Open Source Used In DNAC Platform Arica 2268
Copyright (c) 1999 Computer Systems and Communication Lab, Institute of Information Science, Academia Sinica. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Computer Systems and Communication Lab nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
#
# --------------COPYING.libtabe-----END--------------------------------
#
# --------------COPYING.ipadic-----BEGIN--------------------------------
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# Copyright (c) 2013 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
# (copied below)
# This file is derived from the above dictionary, with slight
# modifications.
# ---------------------------------------------------------------------------
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification,
# are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in
# binary form must reproduce the above copyright notice, this list of
# conditions and the following disclaimer in the documentation and/or
# other materials provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
# ---------------------------------------------------------------------------

4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
# github.com/kanyawtech/myanmar-karen-word-lists
#
# ---------------------------------------------------------------------------
# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

# 7. Database Ownership

# The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.
6. Google double-conversion

Copyright 2006-2011, the V8 project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

UIMA Annotator: Tagger
Copyright 2006-2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

/*
 *
 *       Apache License
 *     Version 2.0, January 2004
 *     http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction,
 * and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by
 * the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not...
pertaining to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
your Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
* incidental, or consequential damages of any character arising as a
* result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill,
* work stoppage, computer failure or malfunction, or any and all
* other commercial damages or losses), even if such Contributor
* has been advised of the possibility of such damages.
*
* 9. Accepting Warranty or Additional Liability. While redistributing
* the Work or Derivative Works thereof, You may choose to offer,
* and charge a fee for, acceptance of support, warranty, indemnity,
* or other liability obligations and/or rights consistent with this
* License. However, in accepting such obligations, You may act only
* on Your own behalf and on Your sole responsibility, not on behalf
* of any other Contributor, and only if You agree to indemnify,
* defend, and hold each Contributor harmless for any liability
* incurred by, or claims asserted against, such Contributor by reason
* of your accepting any such warranty or additional liability.
*
* END OF TERMS AND CONDITIONS
*
* APPENDIX: How to apply the Apache License to your work.
*
* To apply the Apache License to your work, attach the following
* boilerplate notice, with the fields enclosed by brackets "[]"
* replaced with your own identifying information. (Don't include
* the brackets!) The text should be enclosed in the appropriate
* comment syntax for the file format. We also recommend that a
* file or class name and description of purpose be included on the
* same "printed page" as the copyright notice for easier
* identification within third-party archives.
*
* Copyright [yyyy] [name of copyright owner]
*
* Licensed under the Apache License, Version 2.0 (the "License);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231
This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about
this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2016 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
ICU4J (under lucene/analysis/icu) is licensed under an MIT style license (modules/analysis/icu/lib/icu4j-LICENSE-BSD_LIKE.txt) and Copyright (c) 1995-2012 International Business Machines Corporation and others
[The "BSD license"]
Copyright (c) 2015 Terence Parr, Sam Harwell
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement You may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Common Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS
"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS
A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively “Losses”) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay
any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2012 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.
All trademarks and registered trademarks mentioned herein are the property of their respective owners.

# about.md file

## About This Content

May 22, 2015

### License

The Eclipse Foundation makes available all content in this plug-in ("Content"). Unless otherwise indicated below, the Content is provided to you under the terms and conditions of the Apache License, Version 2.0. A copy of the Apache License, Version 2.0 is available at [http://www.apache.org/licenses/LICENSE-2.0.txt](http://www.apache.org/licenses/LICENSE-2.0.txt)

If you did not receive this Content directly from the Eclipse Foundation, the Content is being redistributed by another party ("Redistributor") and different terms and conditions may apply to your use of any object code in the Content. Check the Redistributors license that was provided with the Content. If no such license exists, contact the Redistributor. Unless otherwise indicated below, the terms and conditions of the Apache License, Version 2.0 still apply to any source code in the Content and such source code may be obtained at [http://www.eclipse.org](http://www.eclipse.org).

# notice.md file

Note: the below Eclipse user agreement is standard. It says "Unless otherwise indicated, "... before referring to the EPL. We indicate above that all content is licensed under the ASLv2 license. -- David Smiley

## Eclipse Foundation Software User Agreement

April 9, 2014

### Usage Of Content

THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT AND THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES
### Applicable Licenses

Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is provided with this Content and is also available at [http://www.eclipse.org/legal/epl-v10.html](http://www.eclipse.org/legal/epl-v10.html). For purposes of the EPL, "Program" will mean the Content.

Content includes, but is not limited to, source code, object code, documentation and other files maintained in the Eclipse Foundation source code repository ("Repository") in software modules ("Modules") and made available as downloadable archives ("Downloads").

* Content may be structured and packaged into modules to facilitate delivering, extending, and upgrading the Content.
  * Typical modules may include plug-ins ("Plug-ins"), plug-in fragments ("Fragments"), and features ("Features").
  * Each Plug-in or Fragment may be packaged as a sub-directory or JAR (Java ARchive) in a directory named "plugins".
  * A Feature is a bundle of one or more Plug-ins and/or Fragments and associated material. Each Feature may be packaged as a sub-directory in a directory named "features". Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of the Plug-ins and/or Fragments associated with that Feature.
  * Features may also include other Features ("Included Features"). Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of Included Features.

The terms and conditions governing Plug-ins and Fragments should be contained in files named "about.html" ("Abouts"). The terms and conditions governing Features and Included Features should be contained in files named "license.html" ("Feature Licenses"). Abouts and Feature Licenses may be located in any directory of a Download or Module including, but not limited to the following locations:

* The top-level (root) directory
* Plug-in and Fragment directories
* Inside Plug-ins and Fragments packaged as JARs
* Sub-directories of the directory named "src" of certain Plug-ins
* Feature directories

Note: if a Feature made available by the Eclipse Foundation is installed using the Provisioning Technology (as defined below), you must agree to a license ("Feature Update License") during the installation process. If the Feature contains Included Features, the Feature Update License should either provide you with the terms and conditions governing
Included Features or inform you where you can locate them. Feature Update Licenses may be found in the "license" property of files named "feature.properties" found within a Feature. Such Abouts, Feature Licenses, and Feature Update Licenses contain the terms and conditions (or references to such terms and conditions) that govern your use of the associated Content in that directory.

THE ABOUTS, FEATURE LICENSES, AND FEATURE UPDATE LICENSES MAY REFER TO THE EPL OR OTHER LICENSE AGREEMENTS, NOTICES OR TERMS AND CONDITIONS. SOME OF THESE OTHER LICENSE AGREEMENTS MAY INCLUDE (BUT ARE NOT LIMITED TO):

* Eclipse Distribution License Version 1.0 (available at [http://www.eclipse.org/licenses/edl-v1.0.html](http://www.eclipse.org/licenses/edl-v1.0.html))
* Apache Software License 1.1 (available at [http://www.apache.org/licenses/LICENSE](http://www.apache.org/licenses/LICENSE))
* Apache Software License 2.0 (available at [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0))

IT IS YOUR OBLIGATION TO READ AND ACCEPT ALL SUCH TERMS AND CONDITIONS PRIOR TO USE OF THE CONTENT. If no About, Feature License, or Feature Update License is provided, please contact the Eclipse Foundation to determine what terms and conditions govern that particular Content.

### Use of Provisioning Technology

The Eclipse Foundation makes available provisioning software, examples of which include, but are not limited to, p2 and the Eclipse Update Manager ("Provisioning Technology") for the purpose of allowing users to install software, documentation, information and/or other materials (collectively "Installable Software"). This capability is provided with the intent of allowing such users to install, extend and update Eclipse-based products. Information about packaging Installable Software is available at [http://eclipse.org/equinox/p2/repository_packaging.html](http://eclipse.org/equinox/p2/repository_packaging.html) ("Specification").

You may use Provisioning Technology to allow other parties to install Installable Software. You shall be responsible for enabling the applicable license agreements relating to the Installable Software to be presented to, and accepted by, the users of the Provisioning Technology in accordance with the Specification. By using Provisioning Technology in such a manner and making it available in accordance with the Specification, you further acknowledge your agreement to, and the
acquisition of all necessary rights to permit the following:

1. A series of actions may occur ("Provisioning Process") in which a user may execute the Provisioning Technology on a machine ("Target Machine") with the intent of installing, extending or updating the functionality of an Eclipse-based product.
2. During the Provisioning Process, the Provisioning Technology may cause third party Installable Software or a portion thereof to be accessed and copied to the Target Machine.
3. Pursuant to the Specification, you will provide to the user the terms and conditions that govern the use of the Installable Software ("Installable Software Agreement") and such Installable Software Agreement shall be accessed from the Target Machine in accordance with the Specification. Such Installable Software Agreement must inform the user of the terms and conditions that govern the Installable Software and must solicit acceptance by the end user in the manner prescribed in such Installable Software Agreement. Upon such indication of agreement by the user, the provisioning Technology will complete installation of the Installable Software.

### Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

<small>Java and all Java-based trademarks are trademarks of Oracle Corporation in the United States, other countries, or both.</small>

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.
3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the
terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara.
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licenser for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licenser or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licenser for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was derived from Python 2.4.2 sources available at http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was derived from Python 3.1.2 sources available at http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in core/src/java/org/apache/lucene/util/automaton was derived from Brics automaton sources available at www.brics.dk/automaton/. Here is the copyright from those sources:

/*
 * Copyright (c) 2001-2009 Anders Moeller
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*    derived from this software without specific prior written permission.
*  
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

The levenshtein automata tables in core/src/java/org/apache/lucene/util/automaton
were automatically generated with the moman/finenight FSA package.

Here is the copyright for those sources:

# Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
# Permission is hereby granted, free of charge, to any person
# obtaining a copy of this software and associated documentation
# files (the "Software"), to deal in the Software without
# restriction, including without limitation the rights to use,
# copy, modify, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following
# conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
# OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
# OTHER DEALINGS IN THE SOFTWARE.
Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
* Copyright (C) 1999-2010, International Business Machines
* Corporation and others. All Rights Reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, and/or sell copies of the
* Software, and to permit persons to whom the Software is furnished to do so,
* provided that the above copyright notice(s) and this permission notice appear
* in all copies of the Software and that both the above copyright notice(s) and
* this permission notice appear in supporting documentation.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
* IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
* LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
* ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
* IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION. ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* Except as contained in this notice, the name of a copyright holder shall not
* be used in advertising or otherwise to promote the sale, use or other
* dealings in this Software without prior written authorization of the
* copyright holder.
*/

The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
* this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* Neither the name of the copyright holders nor the names of its contributors
* may be used to endorse or promote products derived from this software
* without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. The names "Center for Intelligent Information Retrieval" and
   "University of Massachusetts" must not be used to endorse or promote products
   derived from this software without prior written permission. To obtain
   permission, contact info@ciir.cs.umass.edu.

THIS SOFTWARE IS PROVIDED BY UNIVERSITY OF MASSACHUSETTS AND OTHER CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

The dictionary comes from Morfologik project. Morfologik uses data from
Polish ispell/myspell dictionary hosted at http://www.sjp.pl/slownik/en/ and
is licenced on the terms of (inter alia) LGPL and Creative Commons
ShareAlike. The part-of-speech tags were added in Morfologik project and
are not found in the data from sjp.pl. The tagset is similar to IPI PAN
tagset.

---

The following license applies to the Morfeusz project,
used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/
Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski,
Marcin Woliski, Robert Woosz

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY COPYRIGHT HOLDERS AS IS AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Morfologik

VERSION: 2.1 PoliMorf
BUILD: 2016-02-13 19:37:50+01:00
GIT: 6e63b53

Copyright (c) 2016, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Ivy (TM)
Copyright 2007-2013 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of Ivy were originally developed by
Jayasoft SARL (http://www.jayasoft.fr/)
and are licensed to the Apache Software Foundation under the
"Software Grant License Agreement"

SSH and SFTP support is provided by the JCraft JSch package,
which is open source software, available under
the terms of a BSD style license.
The original software and related information is available

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

UIMA Base: uimaj-core
Copyright 2006-2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Portions of Apache UIMA were originally developed by International Business Machines Corporation and are licensed to the Apache Software Foundation under the "Software Grant License Agreement", informally known as the "IBM UIMA License Agreement".

Copyright (c) 2003, 2006 IBM Corporation.

javax.servlet-*.jar is under the CDDL license, the original source code for this can be found at http://www.eclipse.org/jetty/downloads.php

Apache Commons Compress
Copyright 2002-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
morfologik-ukrainian-search is a POS tag dictionary in morfologik format adjusted for searching.
It's part of dict_uk project (https://github.com/brown-uk/dict_uk)

Note: to better fit into full-text search model this dictionary has all word forms in lower case but keeps lemmas for proper nouns in upper case.

Licensed under Apache License 2.0.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICEx file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls.
See http://www.jcip.net and the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)

BSD-licensed dictionary of Polish (Morfologik)

VERSION: 2.1 PoliMorf
BUILD: 2016-02-13 19:37:50+01:00
GIT: 6e63b53

Copyright (c) 2016, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2004-2008 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Commons Codec
Copyright 2002-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).


Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org). Verbatim copying and distribution of this entire article is permitted in any medium, provided this notice is preserved.

Apache Commons Logging
Copyright 2003-2013 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Morfologik

VERSION: 2.1 PoliMorf
BUILD: 2016-02-13 19:37:51+01:00
GIT: 6e63b53

Copyright (c) 2016, Marcin Mikowski
Wszelkie prawa zastrzezone

Redystrybucja i uywanie, czy to w formie kodu rdowego, czy w formie kodu wykonawczego, s dozwolone pod warunkiem spenienia ponijszych warunkw:

1. Redystrybucja kodu rdowego musi zawiera powysz not copyrightow, niniejsz list warunkw oraz ponsie owiadczenie o wyczeniu odpowiedzialnoci.
2. Redystrybucja kodu wykonawczego musi zawiera powysz not copyrightow, niniejsz list warunkw oraz ponsie owiadczenie o wyczeniu odpowiedzialnoci w dokumentacji i lub w innych materiaach dostarczanych
TO OPROGRAMOWANIE JEST DOSTARCZONE PRZEZ <POSIADACZA PRAW AUTORSKICH> TAKIM, JAKIE JEST. KADA, DOROZUMIANA LUB BEZPOREDNIO WYRAONA GWARANCJA, NIE WYCZAJC DOROZUMIANEJ GWARANCJI PRZYDATNOCI HANDLOWEJ I PRZYDATNOCI DO OKRELOWEGO ZASTOSOWANIA, JEST WYCZONA. W ADNYM WYPADKU <POSIADACZE PRAW AUTORSKICH> NIE MOG BY ODPOWIEDZIALNI ZA JAKIEKOLWIEK BEZPOREDNIE, POREDNIE, INCYDENTALNE, SPECJALNE, UBOCZNE I WTRNE SZKODY (NIE WYCZAJC OBOWIZKU DOSTARCZENIA PRODUKTU ZASTPCZEGO LUB SERWISU, ODPOWIEDZIALNOCI Z TYTUU UTRATY WALORW UYTKOWYCH, UTRATY DANYCH LUB KORZYCI, A TAKE PRZERW W PRACY PRZEDSIBIORSTWA) SPOWOWANE W JAKIKOLWIEK SPOSW I NA PODSTAWIE ISTNIEJCEJ W TEORII ODPOWIEDZIALNOCI KONTRAKTOWEJ, CAKOWITEJ LUB DELIKTOWEJ (WYNIJEJ ZARWNO Z NIEDBALSTWA JAK INNYCH POSTACI WINY), POWSTAE W JAKIKOLWIEK SPOB W WYNIKU UYWANIA LUB MAJCE ZWIZEK Z UYWANIEM OPROGRAMOWANIA, NAWET JELO MOLIWOCI POWSTANIA TAKICH SZKD OSTRZEENO.

========================================================================= == NOTICE file corresponding to the section 4 d of == == the Apache License, Version 2.0, == == in this case for the Apache Ant distribution. ==
=========================================================================  

Apache Ant  
Copyright 1999-2008 The Apache Software Foundation

This product includes software developed by  
The Apache Software Foundation (http://www.apache.org/).

This product includes also software developed by :  
- the W3C consortium (http://www.w3c.org)  
- the SAX project (http://www.saxproject.org)

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

Portions of this software were originally based on the following:  
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

Apache OpenNLP Maxent  
Copyright 2013 The Apache Software Foundation

This product includes software developed at  
The Apache Software Foundation (http://www.apache.org/).
UIMA Annotator: WhitespaceTokenizer
Copyright 2006-2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache OpenNLP Tools
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects, including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such as the Unicode Character Database. See http://unicode.org/copyright.html for more details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were automatically generated with the moman/finenight FSA library, created by Jean-Philippe Barrette-LaPierre. This library is available under an MIT license, see http://sites.google.com/site/rrettesite/moman and http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
g Package (jaspell): http://jaspell.sourceforge.net/
License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in
  analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
  analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from
  http://snowball.tartarus.org/

The KStem stemmer in
  analysis/common/src/org/apache/lucene/analysis/en
was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst)
under the BSD-license.

The Arabic,Persian,Romanian,Bulgarian, Hindi and Bengali analyzers (common) come with a default
stopword list that is BSD-licensed created by Jacques Savoy.  These files reside in:
  analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt,
  analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt,
  analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt,
  analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt,
  analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt,
  analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt

The German,Spanish,Finnish,French,Hungarian,Italian,Portuguese,Russian and Swedish light stemmers
(common) are based on BSD-licensed reference implementations created by Jacques Savoy and
Ljiljana Dolamic. These files reside in:
  analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
  analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed
by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil,
and Edmond Nolan.
The Polish analyzer (stempel) comes with a default
stopword list that is BSD-licensed created by the Carrot2 project. The file resides
in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt.
See http://project.carrot2.org/license.html.

The SmartChineseAnalyzer source code (smartcn) was
provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_.java (under modules/analysis/common/src/test/)
is derived from Unicode data such as the Unicode Character Database.
See http://unicode.org/copyright.html for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software
developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary
(http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia)
LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP)
(http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original
source code for this can be found at http://www.eclipse.org/jetty/downloads.php

===========================================================================
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

===========================================================================
mecab-ipadic-2.7.0-20070801 Notice
===========================================================================

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
IMPLIED, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

Nori Korean Morphological Analyzer - Apache Lucene Integration

This software includes a binary and/or source version of data from

mecab-ko-dic-2.0.3-20170922

which can be obtained from

https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz
SourceFileDumpArchiveEntry.java::z||Forg/apache/commons/compress/archivers/dump/DumpArchiveEntry$PERMISSION@|E~89java/util/HashSetEU<=>yEF),...,0,1,2,3,4,5,6,7,java/lang/Enum
java/util/Setclone()Ljava/lang/Object;|oracle/apache/commons/compress/archivers/dump/DumpArchiveEntry5(Ljava/lang/Class;Ljava/lang/String;)Ljava/lang/Enum(I)Vadd(Ljava/lang/Object;)ZisEmpty()Zjava/util/CollectionemptySet()Ljava/util/Set;java/util/EnumSetcopyOf+(Ljava/util/Collection;)Ljava/util/EnumSet;@1(@),@-@..@/,@0,1,2,3,4,5,6,7,89;::<=>
?%@A>4
*%B
CDEF>H*+*?567BG,89HIJK>NYL
M,>6,2:-->W+
+?":<=->6<<BECIFB L,N89FMNOFMPQR#HSTU}>YYYYYY@Y YY !Y"#Y$
%Y&'YSYSYSYSYSYSY!SY#SY
%SY'S?6
&'0)@*P+_.-../01%HVWX+
\*@%

This product includes data from BSD-licensed dictionary of Polish (Morfologik, PoliMorf) (http://morfologik.blogspot.com/)
CyberNeko HTML Parser
(C) Copyright 2002-2009, Andy Clark, Marc Guillemot. All rights reserved.

NekoHTML is a simple HTML scanner and tag balancer that enables application programmers to parse HTML documents and access the information using standard XML interfaces. The parser can scan HTML files and "fix up" many common mistakes that human (and computer) authors make in writing HTML documents. NekoHTML adds missing parent elements; automatically closes elements with optional end tags; and can handle mismatched inline element tags.

NekoHTML is written using the Xerces Native Interface (XNI) that is the foundation of the Xerces2 implementation. This enables you to use the NekoHTML parser with existing XNI tools without modification or rewriting code.
Copyright (c) 2012 France Tlcom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

------------------------------------------------------------------------------
License for JCraft JSch package
------------------------------------------------------------------------------
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------------------------
License for jQuery
------------------------------------------------------------------------------
Copyright (c) 2007 John Resig, http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

Open Source Used In DNAC Platform Arica 2336
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=========================================================================  ==  SLF4J Notice -- http://www.slf4j.org/license.html ==  ==  =========================================================================

Copyright (c) 2004-2008 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache HttpComponents Client
Copyright 1999-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls. See http://www.jcip.net
This product includes BSD-licensed software developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).
JUnit (under lib/junit-4.10.jar) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html
RandomizedRunner, a JUnit @Runner for randomized tests (and more)
Copyright 2011-2012 Carrot Search s.c.
http://labs.carrotsearch.com/randomizedtesting.html

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product includes asm (asmlib), BSD license
This product includes Google Guava, ASL license
This product includes simple-xml, ASL license
This product includes Google GSON, ASL license

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.622 string-prototype-trimstart 1.0.1

1.622.1 Available under license :

MIT License

Copyright (c) 2017 Khaled Al-Ansari

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.623 tephra-hbase-compat-1.0 0.6.0

1.623.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright © 2012-2014 Cask Data, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/coprocessor/TransactionProcessor.java
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/Filters.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright © 2014 Cask Data, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/TransactionAwareHTable.java
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/SecondaryIndexTable.java
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/coprocessor/TransactionVisibilityFilter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright © 2015 Cask Data, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1120497655_1609977972.89/0/tephra-hbase-compat-1-0-0-6-0-sources-1-
jar/co/cask/tephra/hbase10/HBase10ConfigurationProvider.java

1.624 jansi 1.8
1.624.1 Available under license :
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2009, Progress Software Corporation and/or its
 * subsidiaries or affiliates. All rights reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
Found in path(s):
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/WindowedAnsiOutputStream.java
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/AnsiOutputStream.java
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/Ansi.java
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/AnsiConsole.java

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/AnsiString.java
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/AnsiRenderWriter.java
* /opt/cola/permits/1257860190_1643019494.64/0/jansi-1-8-sources-jar/org/fusesource/jansi/AnsiRenderer.java

1.625 hk2-api 2.5.0-b32
1.625.1 Available under license:
CDDL + GPLv2 with classpath exception
https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
A business-friendly OSS license

1.626 serve-static 1.13.2 1.13.2
1.626.1 Available under license:
(The MIT License)
Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.627 cliui 3.2.0 3.2.0
1.627.1 Available under license :
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software
for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice
appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE
LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.628 ms 0.7.1 0.7.1
1.628.1 Available under license :
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <rauchg@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.629 lucene-sandbox 7.4.0
1.629.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from unicode conversion examples available at
http://www.unicode.org/Public/PROGRAMS/CVTUTF. Here is the copyright
from those sources:

/*
 * Copyright 2001-2004 Unicode, Inc.
 * * Disclaimer
 * * This source code is provided as is by Unicode, Inc. No claims are
 * made as to fitness for any particular purpose. No warranties of any
 * kind are expressed or implied. The recipient agrees to determine
* applicability of information provided. If this file has been
* purchased on magnetic or optical media from Unicode, Inc., the
* sole remedy for any claim will be exchange of defective media
* within 90 days of receipt.
*
* Limitations on Rights to Redistribute This Code
*
* Unicode, Inc. hereby grants the right to freely use the information
* supplied in this file in the creation of products supporting the
* Unicode Standard, and to make copies of this file in any form
* for internal or external distribution as long as this notice
* remains attached.
*/

Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was
derived from Python 2.4.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from Python 3.1.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in core/src/java/org/apache/lucene/util/automaton was
derived from Brics automaton sources available at
www.brics.dk/automaton/. Here is the copyright from those sources:

/*
* Copyright (c) 2001-2009 Anders Moeller
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
*/
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

The levenshtein automata tables in core/src/java/org/apache/lucene/util/automaton
were automatically generated with the moman/finenight FSA package.
Here is the copyright for those sources:

```
# Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
# Permission is hereby granted, free of charge, to any person
# obtaining a copy of this software and associated documentation
# files (the "Software"), to deal in the Software without
# restriction, including without limitation the rights to use,
# copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following
# conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
# OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
# OTHER DEALINGS IN THE SOFTWARE.
```

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
 * Copyright (C) 1999-2010, International Business Machines
 * Corporation and others. All Rights Reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, and/or sell copies of the
* Software, and to permit persons to whom the Software is furnished to do so,
* provided that the above copyright notice(s) and this permission notice appear
* in all copies of the Software and that both the above copyright notice(s) and
* this permission notice appear in supporting documentation.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
* IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
* LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
* ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
* IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* Except as contained in this notice, the name of a copyright holder shall not
* be used in advertising or otherwise to promote the sale, use or other
* dealings in this Software without prior written authorization of the
* copyright holder.
*/

The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
* this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* Neither the name of the copyright holders nor the names of its contributors
* may be used to endorse or promote products derived from this software
* without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names “Center for Intelligent Information Retrieval” and “University of Massachusetts” must not be used to endorse or promote products derived from this software without prior written permission. To obtain permission, contact info@ciir.cs.umass.edu.

THE SOFTWARE IS PROVIDED BY UNIVERSITY OF MASSACHUSETTS AND OTHER CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

The dictionary comes from Morfologik project. Morfologik uses data from Polish ispell/myspell dictionary hosted at http://www.sjp.pl/slownik/en/ and is licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike. The part-of-speech tags were added in Morfologik project and are not found in the data from sjp.pl. The tagset is similar to IPI PAN tagset.

---

The following license applies to the Morfeusz project, used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/

Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski, Marcin Woliski, Robert Woosz

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY COPYRIGHT HOLDERS AS IS AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects,
including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license
and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such
as the Unicode Character Database. See http://unicode.org/copyright.html for more
details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is
BSD-licensed, created by Anders Müller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were
automatically generated with the moman/finenight FSA library, created by
Jean-Philippe Barrette-LaPierre. This library is available under an MIT license,
see http://sites.google.com/site/rettyesite/moman and
http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from
the Apache CXF project and is Apache License 2.0.
The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
g Package (jaspell): http://jaspell.sourceforge.net/
License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in
   analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
   analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from
   http://snowball.tartarus.org/

The KStem stemmer in
   analysis/common/src/org/apache/lucene/analysis/en
was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst)
under the BSD-license.

The Arabic,Persian,Romanian,Bulgarian, Hindi and Bengali analyzers (common) come with a default
stopword list that is BSD-licensed created by Jacques Savoy. These files reside in:
analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt

The German,Spanish,Finnish,French,Hungarian,Italian,Portuguese,Russian and Swedish light stemmers
(common) are based on BSD-licensed reference implementations created by Jacques Savoy and
Ljiljana Dolamic. These files reside in:
analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fi/FinnishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
The Stempel analyzer (stempel) includes BSD-licensed software developed by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil, and Edmond Nolan.

The Polish analyzer (stempel) comes with a default stopword list that is BSD-licensed created by the Carrot2 project. The file resides in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt. See http://project.carrot2.org/license.html.

The SmartChineseAnalyzer source code (smarten) was provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/) is derived from Unicode data such as the Unicode Character Database. See http://unicode.org/copyright.html for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary (http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP) (http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-.jar are under the CDDL license, the original source code for this can be found at http://www.eclipse.org/jetty/downloads.php

===========================================================================
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from
mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

===========================================================================

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortious action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
implied, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term “use” as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

===========================================================================
Nori Korean Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ko-dic-2.0.3-20170922

which can be obtained from

https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz

1.630 json5 2.2.0
1.630.1 Available under license : MIT License

Copyright (c) 2012-2018 Aseem Kishore, and [others].

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[others]: https://github.com/json5/json5/contributors

1.631 es5-ext 0.10.53
1.631.1 Available under license :
ISC License

Copyright (c) 2011-2019, Mariusz Nowak, @medikoo, medikoo.com

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.632 commons-digester 1.8

1.633 commons-cli 1.2
1.633.1 Available under license :
Apache Commons CLI
Copyright 2001-2009 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.634 pinkie 2.0.4 2.0.4
1.634.1 Available under license :
   The MIT License (MIT)

   Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)

   Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software,

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.635 lodash.isnil 4.0.0 4.0.0
1.635.1 Available under license :
   Copyright 2012-2016 The Dojo Foundation <http://dojofoundation.org/>
Based on Underscore.js, copyright 2009-2016 Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

   Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

   The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.636 callsites 0.2.0

1.636.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.637 intl-messageformat-parser 1.4.0 1.4.0

1.637.1 Available under license:
Copyright 2014 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------

Inspired by and derived from:
messageformat.js https://github.com/SlexAxton/messageformat.js
Copyright 2014 Alex Sexton
Apache License, Version 2.0

1.638 st4 4.1

1.638.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * [The "BSD license"]
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

tree grammar CodeGenerator;

options {
  language=Java;
  tokenVocab=STParser;
  ASTLabelType=CommonTree;
}

@header {
  /*
  * [The "BSD license"]
  * Copyright (c) 2011 Terence Parr
  * All rights reserved.
  *
  * Redistribution and use in source and binary forms, with or without
  * modification, are permitted provided that the following conditions
  * are met:
  * 1. Redistributions of source code must retain the above copyright
     notice, this list of conditions and the following disclaimer.
  * 2. Redistributions in binary form must reproduce the above copyright
     notice, this list of conditions and the following disclaimer in the
     documentation and/or other materials provided with the distribution.
  * 3. The name of the author may not be used to endorse or promote products
     derived from this software without specific prior written permission.
     *
  * THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
  * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
  * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
  * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
  * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
  * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
  * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
  * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
  * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  */
}
package org.stringtemplate.v4.compiler;
import org.stringtemplate.v4.misc.*/
import org.stringtemplate.v4.*;
{

@members {
String outermostTemplateName:// name of overall template
CompiledST outermostImpl;
Token templateToken;// overall template token
String template;  // overall template text
ErrorManager errMgr;
public CodeGenerator(TreeNodeStream input, ErrorManager errMgr, String name, String template, Token templateToken) {
this(input, new RecognizerSharedState());
this.errMgr = errMgr;
this.outermostTemplateName = name;
this.template = template;
this.templateToken = templateToken;
}

public void addArgument(List<FormalArgument> args, Token t) {
String name = t.getText();
for (FormalArgument arg : args) {
if (arg.name.equals(name)) {
errMgr.compileTimeError(ErrorType.PARAMETER_REDEFINITION, templateToken, t, name);
return;
}
}

args.add(new FormalArgument(name));
}

// convience funcs to hide offensive sending of emit messages to
// CompilationState temp data object.

public void emit1(CommonTree opAST, short opcode, int arg) {
$template::state.emit1(opAST, opcode, arg);
}
public void emit1(CommonTree opAST, short opcode, String arg) {
$template::state.emit1(opAST, opcode, arg);
}
public void emit2(CommonTree opAST, short opcode, int arg, int arg2) {
$template::state.emit2(opAST, opcode, arg, arg2);
}
public void emit2(CommonTree opAST, short opcode, String s, int arg2) {
$template::state.emit2(opAST, opcode, s, arg2);
}
```
public void emit(CommonTree opAST, short opcode) {
    $template::state.emit(opAST, opcode);
}
public void insert(int addr, short opcode, String s) {
    $template::state.insert(addr, opcode, s);
}
public void setOption(CommonTree id) {
    $template::state.setOption(id);
}
public void write(int addr, short value) {
    $template::state.write(addr, value);
}
public int address() { return $template::state.ip; }
public void func(CommonTree id) { $template::state.func(templateToken, id); }
public void refAttr(CommonTree id) { $template::state.refAttr(templateToken, id); }
public int defineString(String s) { return $template::state.defineString(s); }

@Override
public void displayRecognitionError(String[] tokenNames, RecognitionException e) {
    Token tokenWithPosition = e.token;
    if (tokenWithPosition.getInputStream() == null) {
        tokenWithPosition = input.getTreeAdaptor().getToken(input.LT(-1));
    }
    String hdr = getErrorHeader(e);
    String msg = getErrorMessage(e, tokenNames);
    errMgr.compileTimeError(ErrorType.SYNTAX_ERROR, templateToken, tokenWithPosition, hdr + " " + msg);
}

templateAndEOF : template[null,null] EOF; // hush warning; ignore

template[String name, List<FormalArgument> args] returns [CompiledST impl]
scope {
    CompilationState state; // automatically get a new state pointer per invocation
}
@init {
$template::state = new CompilationState(errMgr, name, input.getTokenStream());
$impl = $template::state.impl;
if ( $template.size() == 1 ) outermostImpl = $impl;
$impl.defineFormalArgs($args); // make sure args are defined prior to compilation
if ( name!=null && name.startsWith(Compiler.SUBTEMPLATE_PREFIX) ) {
    $impl.addArg(new FormalArgument("i"));
    $impl.addArg(new FormalArgument("i0"));
} $impl.template = template; // always forget the entire template; char indexes are relative to it
}
:chunk
```
{ // finish off the CompiledST result
    if ( $template::state.stringtable!=null ) $impl.strings = $template::state.stringtable.toArray();
    $impl.codeSize = $template::state.ip;
}

chunk
element*

;

element
:^(INDENTED_EXPR INDENT compoundElement[$INDENT]) // ignore indent in front of IF and region blocks
    [compoundElement>null]
|^(INDENTED_EXPR INDENT {$template::state.indent($INDENT);} singleElement?
    {$template::state.emit(Bytecode.INSTR_DEDENT);})
|singleElement

;

singleElement
e:exprElement
|TEXT
|
if ( $TEXT.text.length()>0 ) {
    emit1($TEXT,Bytecode.INSTR_WRITE_STR, $TEXT.text);
}

[NEWLINE {emit($NEWLINE, Bytecode.INSTR_NEWLINE);}]

;

compoundElement[CommonTree indent]
  :ifstat[indent]
  |region[indent]

;

e:exprElement
  @init { short op = Bytecode.INSTR_WRITE; }
  :^( Expr expr (exprOptions {op=Bytecode.INSTR_WRITE_OPT;})? )
|
/*
CompilationState state = $template::state;
CompiledST impl = state.impl;
if ( impl.instrs[state.ip-1] == Bytecode.INSTR_LOAD_LOCAL ) {
    impl.instrs[state.ip-1] = Bytecode.INSTR_WRITE_LOCAL;
} else {
    emit($EXPR, op);
}
/*
emit($EXPR, op);
}
;

region[CommonTree indent] returns [String name]
@init {
if ( indent!=null ) $template::state.indent(indent);
}
@after {
if ( indent!=null ) $template::state.emit(Bytecode.INSTR_DEDENT);
}
:^REGION ID
{$name = STGroup.getMangledRegionName(outermostTemplateName, $ID.text);
} template[$name,null]
{
CompiledST sub = $template.impl;
   sub.isRegion = true;
   sub.regionDefType = ST.RegionType.EMBEDDED;
   sub.templateDefStartToken = $ID.token;
//sub.dump();
outermostImpl.addImplicitlyDefinedTemplate(sub);
emit2($start, Bytecode.INSTR_NEW, $region.name, 0);
emit($start, Bytecode.INSTR_WRITE);
}
);

subtemplate returns [String name, int nargs]
@init {
   $name = Compiler.getNewSubtemplateName();
List<FormalArgument> args = new ArrayList<FormalArgument>();
}
:^SUBTEMPLATE
(^ARGS (ID {addArgument(args, $ID.token);})+))
{$nargs = args.size();}
template[$name,args]
{
CompiledST sub = $template.impl;
sub.isAnonSubtemplate = true;
   sub.templateDefStartToken = $SUBTEMPLATE.token;
   sub.ast = $SUBTEMPLATE;
   sub.ast.setUnknownTokenBoundaries();
   sub.tokens = input.getTokenStream();
//sub.dump();
outermostImpl.addImplicitlyDefinedTemplate(sub);
}
<table>
<thead>
<tr>
<th>SUBTEMPLATE // {}</th>
</tr>
</thead>
<tbody>
<tr>
<td>CompiledST sub = new CompiledST();</td>
</tr>
<tr>
<td>sub.name = $name;</td>
</tr>
<tr>
<td>sub.template = &quot;&quot;;</td>
</tr>
<tr>
<td>sub.addArg(new FormalArgument(&quot;i&quot;));</td>
</tr>
<tr>
<td>sub.addArg(new FormalArgument(&quot;i0&quot;));</td>
</tr>
<tr>
<td>sub.isAnonSubtemplate = true;</td>
</tr>
<tr>
<td>sub.templateDefStartToken = $SUBTEMPLATE.token;</td>
</tr>
<tr>
<td>sub.ast = $SUBTEMPLATE;</td>
</tr>
<tr>
<td>sub.ast.setUnknownTokenBoundaries();</td>
</tr>
<tr>
<td>sub.tokens = input.getTokenStream();</td>
</tr>
</tbody>
</table>
//sub.dump(); |
outermostImpl.addImplicitlyDefinedTemplate(sub); |
|
@init { |
/** Tracks address of branch operand (in code block). It's how |
* we backpatch forward references when generating code for IFs. |
*/ |
int prevBranchOperand = -1; |
/** Branch instruction operands that are forward refs to end of IF. |
* We need to update them once we see the endif. |
*/ |
List<Integer> endRefs = new ArrayList<Integer>(); |
if ( indent!=null ) $template::state.indent(indent); |
} |
@after { |
if ( indent!=null ) $template::state.emit(Bytecode.INSTR_DEDENT); |
} |
:_(i='if' conditional |
{ |
prevBranchOperand = address()+1; |
emit1($i,Bytecode.INSTR_BRF, -1); // write placeholder as branch target |
} |
chunk |
__(eif='elseif' |
{ |
endRefs.add(address()+1); |
emit1($eif,Bytecode.INSTR_BR, -1); // br end |
// update previous branch instruction |
write(prevBranchOperand, (short)address()); |
prevBranchOperand = -1; |
} |
} |
ec=conditional |
|
prevBranchOperand = address()+1;
// write placeholder as branch target
emit1($ec.start, Bytecode.INSTR_BRF, -1);
}
chunk
)
)*
(^)(el='else'
{
endRefs.add(address()+1);
emit1($el, Bytecode.INSTR_BR, -1); // br end
// update previous branch instruction
write(prevBranchOperand, (short)address());
prevBranchOperand = -1;
}
chunk
)
)?
)
{
if ( prevBranchOperand>=0 ) {
write(prevBranchOperand, (short)address());
}

for (int opnd : endRefs) write(opnd, (short)address());
}
;
includeExpr

prop:^PROP expr ID)(emit1($PROP, Bytecode.INSTR_LOAD_PROP, $ID.text);
[^PROP_IND expr expr)(emit($PROP_IND, Bytecode.INSTR_LOAD_PROP_IND);

mapTemplateRef[int num_exprs]
:^INCLUDE qualifiedId
{for (int i=1; i<=$num_exprs; i++) emit($INCLUDE,Bytecode.INSTR_NULL);
args
}
if ( $args.passThru ) emit1($start, Bytecode.INSTR_PASSTHRU, $qualifiedId.text);
if ( $args.namedArgs ) emit1($INCLUDE, Bytecode.INSTR_NEW_BOX_ARGS, $qualifiedId.text);
else emit2($INCLUDE, Bytecode.INSTR_NEW, $qualifiedId.text, $args.n+$num_exprs);
}
|subtemplate
{|if ( $subtemplate.nargs != $num_exprs ) {
errMgr.compileTimeError(ErrorType.ANON_ARGUMENT_MISMATCH,
templateToken, $subtemplate.start.token, $subtemplate.nargs, $num_exprs);
}
for (int i=1; i<=$num_exprs; i++) emit($subtemplate.start,Bytecode.INSTR_NULL);
emit2($subtemplate.start, Bytecode.INSTR_NEW,
 $subtemplate.name,
 $num_exprs);
}
[^INCLUDE_IND expr
{|emit($INCLUDE_IND,Bytecode.INSTR_TOSTR);
for (int i=1; i<=$num_exprs; i++) emit($INCLUDE_IND,Bytecode.INSTR_NULL);
}
args
{|emit1($INCLUDE_IND, Bytecode.INSTR_NEW_IND, $args.n+$num_exprs);
}

includeExpr
:^EXEC_FUNC ID expr?)(func($ID);
[^INCLUDE qualifiedId args)
{if ( $args.passThru ) emit1($start, Bytecode.INSTR_PASSTHRU, $qualifiedId.text);
if ( $args.namedArgs ) emit1($INCLUDE, Bytecode.INSTR_NEW_BOX_ARGS, $qualifiedId.text);
else emit2($INCLUDE, Bytecode.INSTR_NEW, $qualifiedId.text, $args.n);
}
|^(INCLUDE_SUPER ID args)
{
if ( $args.passThru ) emit1($start, Bytecode.INSTR_PASSTHRU, $ID.text);
if ( $args.namedArgs ) emit1($INCLUDE_SUPER, Bytecode.INSTR_SUPER_NEW_BOX_ARGS, $ID.text);
else emit2($INCLUDE_SUPER, Bytecode.INSTR_SUPER_NEW, $ID.text, $args.n);
}
|^(INCLUDE_REGION ID){
CompiledST impl =
Compiler.defineBlankRegion(outermostImpl, $ID.token);
//impl.dump();
emit2($INCLUDE_REGION,Bytecode.INSTR_NEW,impl.name,0);
}
|^(INCLUDE_SUPER_REGION ID){
String mangled =
STGroup.getMangledRegionName(outermostImpl.name, $ID.text);
emit2($INCLUDE_SUPER_REGION,Bytecode.INSTR_SUPER_NEW,mangled,0);
}
|primary
;
primary
:ID{refAttr($ID);}
|STRING{emit1($STRING,Bytecode.INSTR_LOAD_STR, Misc.strip($STRING.text,1));}
|TRUE{emit($TRUE, Bytecode.INSTR_TRUE);}
|FALSE{emit($FALSE, Bytecode.INSTR_FALSE);}
|subtemplate// push a subtemplate but ignore args since we can't pass any to it here
{emit2($start,Bytecode.INSTR_NEW, $subtemplate.name, 0);}
|list
|^(INCLUDE_IND
expr {emit($INCLUDE_IND, Bytecode.INSTR_TOSTR);}
args
{emit1($INCLUDE_IND, Bytecode.INSTR_NEW_IND, $args.n);}
)
|^(TO_STR expr){emit($TO_STR, Bytecode.INSTR_TOSTR);}
;
qualifiedId
:^(SLASH qualifiedId ID)
|^(SLASH ID)
|ID
;
arg : expr ;
args returns [int n=0, boolean namedArgs=false, boolean passThru]
:( arg {$n++;} )+
|{emit($args.start, Bytecode.INSTR_ARGS); $namedArgs=true;}
(^(eq='=' ID expr)

Open Source Used In DNAC Platform Arica 2378


open source used in DNAC platform Arica

/*
 * [The "BSD license"]
 * Copyright (c) 2012 Terence Parr
 * Copyright (c) 2012 Sam Harwell
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

No license file was found, but licenses were detected in source scan.

Found in path(s):
*/

/opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/compiler/CodeGenerator.g
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2011 Terence Parr
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/InstanceScope.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/compiler/Compiler.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/debug/ConstructionEvent.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/compiler/CompiledST.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/AutoIndentWriter.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/debug/InterpEvent.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/gui/JTreeSTModel.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/misc/ErrorBuffer.java
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/misc/Misc.java
No license file was found, but licenses were detected in source scan.

/*
 * [The "BSD license"]
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 *    derived from this software without specific prior written permission.
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

grammar Group;

options {
    language=Java;
}

tokens {
    TRUE='true'; FALSE='false'; LBRACK='['; RBRACK=']';
}

@header {

    /**************************************************************************
    * [The "BSD license"]
    * Copyright (c) 2011 Terence Parr
    * All rights reserved.
    *
    * Redistribution and use in source and binary forms, with or without
    * modification, are permitted provided that the following conditions
    * are met:
    * 1. Redistributions of source code must retain the above copyright
       notice, this list of conditions and the following disclaimer.
    * 2. Redistributions in binary form must reproduce the above copyright
       notice, this list of conditions and the following disclaimer in the
       documentation and/or other materials provided with the distribution.
    * 3. The name of the author may not be used to endorse or promote products
       derived from this software without specific prior written permission.
    *
    * THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
    * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
    * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
    * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
    * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
    * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
    * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
    * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
    * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
    * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
    */

    package org.stringtemplate.v4.compiler;
import java.util.Collections;
import java.util.Map;
import java.util.LinkedHashMap;
import java.util.List;
import java.util.ArrayList;
import org.stringtemplate.v4.misc.);
import org.stringtemplate.v4.*;
import java.io.File;
}

package org.stringtemplate.v4.compiler;
import org.stringtemplate.v4.*;
import org.stringtemplate.v4.misc.);
import java.io.File;
}

@members {
 public STGroup group;
}
@Override
public void displayRecognitionError(String[] tokenNames,
RecognitionException e)
{
String msg = getErrorMessage(e, tokenNames);
group.errMgr.groupSyntaxError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);
}
@Override
public String getSourceName() {
String fullFileName = super.getSourceName();
File f = new File(fullFileName); // strip to simple name
return f.getName();
}
public void error(String msg) {
NoViableAltException e = new NoViableAltException("", 0, 0, input);
group.errMgr.groupSyntaxError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);
recover(input, null);
}
public void addArgument(List<FormalArgument> args, Token t, Token defaultValueToken) {
String name = t.getText();
for (FormalArgument arg : args) {
if (arg.name.equals(name)) {
group.errMgr.compileTimeError(ErrorType.PARAMETER_REDEFINITION, null, t, name);
return;
}
}
args.add(new FormalArgument(name, defaultValueToken));
}
}
@lexer::members {
public STGroup group;
@Override
public void reportError(RecognitionException e) {
String msg = null;
if ( e instanceof NoViableAltException ) {
msg = "invalid character '"+(char)input.LA(1)+"'";
}
else if ( e instanceof MismatchedTokenException && ((MismatchedTokenException)e).expecting=='"' ) {
msg = "unterminated string";
}
else {
msg = getErrorMessage(e, getTokenNames());
}
group.errMgr.groupSyntaxError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);

Open Source Used In DNAC Platform Arica 2385


@Override
public String getSourceName() {
    String fullFileName = super.getSourceName();
    File f = new File(fullFileName); // strip to simple name
    return f.getName();
}
}

group[STGroup group, String prefix]
@init {
    GroupLexer lexer = (GroupLexer)input.getTokenSource();
    this.group = lexer.group = $group;
}
:oldStyleHeader?
delimiters?
    ('import' STRING {group.importTemplates($STRING);})
['import' // common error: name not in string
     MismatchedTokenException e = new MismatchedTokenException(STRING, input);
     reportError(e);
    ]
ID ('.' ID)* // might be a.b.c.d
)*
def[prefix]*
EOF
;

oldStyleHeader // ignore but lets us use this parser in AW for both v3 and v4
    : 'group' ID ('.' ID)?
        ('implements' ID ('.' ID)* )?
        '
        ;

groupName returns [String name]
@init {StringBuilder buf = new StringBuilder();}
:a=ID {buf.append($a.text);} ('.' a=ID {buf.append($a.text);})*
;

delimiters
    : 'delimiters' a=STRING ',' b=STRING
    {
        boolean supported = true;
        char startCharacter = $a.getText().charAt(1);
        if (STGroup.isReservedCharacter(startCharacter)) {
            group.errMgr.compileTimeError(ErrorType.UNSUPPORTED_DELIMITER, null, $a,
            String.valueOf(startCharacter));
            supported = false;
        }
    }
char stopCharacter = $b.getText().charAt(1);
if (!STGroup.isReservedCharacter(stopCharacter)) {
    group.errMgr.compileTimeError(ErrorType.UNSUPPORTED_DELIMITER, null, $b,
    String.valueOf(stopCharacter));
    supported = false;
}

if (supported) {
    group.delimiterStartChar=$a.getText().charAt(1);
    group.delimiterStopChar=$b.getText().charAt(1);
}

/** Match template and dictionary defs outside of (...)+ loop in group.
 * The key is catching while still in the loop; must keep prediction of
 * elements separate from "stay in loop" prediction.
 */
def[String prefix] : templateDef[prefix] | dictDef ;
catch[RecognitionException re] {
    // pretend we already saw an error here
    state.lastErrorIndex = input.index();
    error("garbled template definition starting at "+input.LT(1).getText()+"++");
}

templateDef[String prefix]
@init {
    String template=null;
    int n=0; // num char to strip from left, right of template def
}
:'@' enclosing=ID '.' name=ID '(' ')'
|name=ID '(' formalArgs ')'
|'::=
    {Token templateToken = input.LT(1);}  
(STRING   {template=$STRING.text; n=1;}  
   [BIGSTRING   {template=$BIGSTRING.text; n=2;}  
    [BIGSTRING_NO_NL   {template=$BIGSTRING_NO_NL.text; n=2;}  
    |
    template = "";
    String msg = "missing template at "+input.LT(1).getText()+"++";
    NoViableAltException e = new NoViableAltException("", 0, 0, input);
    group.errMgr.groupSyntaxError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);
}
if ( $name.index >= 0 ) { // if ID missing
template = Misc.strip(template, n);
String templateName = $name.text;
if ( prefix.length() > 0 ) templateName = prefix + $name.text;
String enclosingTemplateName = $enclosing.text;
if ( enclosingTemplateName != null && enclosingTemplateName.length() > 0 && prefix.length() > 0 ) {
    enclosingTemplateName = prefix + enclosingTemplateName;
}
group.defineTemplateOrRegion(templateName, enclosingTemplateName, templateToken,
    template, $name, $formalArgs.args);
}
| alias=ID '::=' target=ID  {group.defineTemplateAlias($alias, $target);}
;

formalArgs returns [List<FormalArgument> args = new ArrayList<FormalArgument>()]
scope {
    boolean hasOptionalParameter;
}
@init { $formalArgs::hasOptionalParameter = false; }
:formalArg[$args] (',', formalArg[$args])*
|
;

formalArg[List<FormalArgument> args]
:ID
tilde a=(STRING|ANONYMOUS_TEMPLATE|true|false) {$formalArgs::hasOptionalParameter = true;}
tilde a='[' '] ' {$formalArgs::hasOptionalParameter = true;}
|
if ($formalArgs::hasOptionalParameter) {
group.errMgr.compileTimeError(ErrorType.REQUIRED_PARAMETER_AFTER_OPTIONAL,
    null, $ID);
}
|
{addArgument($args, $ID, $a);}
:

/*
suffix returns [int cardinality=FormalArgument.REQUIRED]
    : OPTIONAL
    | STAR
    | PLUS
    |
    */
dictDef
Open Source Used In DNAC Platform Arica  2389

```java
:ID ':= dict
{
    if ( group.rawGetDictionary($ID.text)!=null ) {
        group.errMgr.compileTimeError(ErrorType.MAP_REDEFINITION, null, $ID);
    } else if ( group.rawGetTemplate($ID.text)!=null ) {
        group.errMgr.compileTimeError(ErrorType.TEMPLATE_REDEFINITION_AS_MAP, null, $ID);
    } else {
        group.defineDictionary($ID.text, $dict.mapping);
    }
}

dict returns [Map<String,Object> mapping]
@init {mapping=new LinkedHashMap<String,Object>();}:
    '[' dictPairs[mapping] ']' ;

dictPairs[Map<String,Object> mapping]
    :keyValuePair[mapping]
        (',' keyValuePair[mapping])* (',' defaultValuePair[mapping])?
    [defaultValuePair[mapping]]
    ;
    catch[RecognitionException re] {
        error("missing dictionary entry at "+input.LT(1).getText()+");
    }

defaultValuePair[Map<String,Object> mapping]
    :default' ':' keyValue {mapping.put(STGroup.DEFAULT_KEY, $keyValue.value);}
    ;

keyValuePair[Map<String,Object> mapping]
    :STRING ':' keyValue {mapping.put(Misc.replaceEscapes(Misc.strip($STRING.text, 1)), $keyValue.value);}
    ;

keyValue returns [Object value]
    :BIGSTRING{$value = group.createSingleton($BIGSTRING);}
    [BIGSTRING_NO_NL{$value = group.createSingleton($BIGSTRING_NO_NL);}
    [ANONYMOUS_TEMPLATE{$value = group.createSingleton($ANONYMOUS_TEMPLATE);}
    [STRING{$value = Misc.replaceEscapes(Misc.strip($STRING.text, 1));}
    [TRUE{$value = true;}
    [FALSE{$value = false;}
    [' '][$value = Collections.emptyList();]
    [catch[RecognitionException re] {
```
error("missing value for key at"+input.LT(1).getText()+");

ID:(a..'z'|A..'Z'|'_') (a..'z'|A..'Z'|0..'9'|'_'-)*
;

STRING :
:";
(\\:" ;
|\\ ~""
|[
String msg = "\n in string";
   NoViableAltException e = new NoViableAltException("", 0, 0, input);
group.errMgr.groupLexerError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);
}
'\n'}
|-(\\""|\\')
)*

   }
   String txt = getText().replaceAll("\\\n","\n");
setText(txt);
}

BIGSTRING_NO_NL // same as BIGSTRING but means ignore newlines later
:<< ( .)* '>>'
   // %\> is the escape to avoid end of string
   }
   String txt = getText().replaceAll("\\%\\>}","\\%>");
setText(txt);
}

/** Match <<...>> but also allow <<..<x>>> so we can have tag on end.
   Escapes: >> means >> inside of <<...>>.
   Escapes: \>> means >> inside of <<...>> unless at end like <<...\>>>.
   In that case, use <%.>>%> instead.
*/
BIGSTRING :
:<<
   (options {greedy=false;}
   :> // > escape
   |\\ ~> // allow this but don't collapse in action
   |
   )*
   '>>>'
   

String txt = getText();
    txt = Misc.replaceEscapedRightAngle(txt); // replace \\ with > unless <\\>
setText(txt);
}
;

ANONYMOUS_TEMPLATE
   :'{'
   {
Token templateToken = new CommonToken(input, ANONYMOUS_TEMPLATE, 0, getCharIndex(),
getCharIndex());
STLexer lexer =
new STLexer(group.errMgr, input, templateToken, group.delimiterStartChar, group.delimiterStopChar);
lexer.subtemplateDepth = 1;
Token t = lexer.nextToken();
while ( lexer.subtemplateDepth>=1 || t.getType()!=STLexer.RCURLY ) {
    if ( t.getType()==STLexer.EOF_TYPE ) {
        MismatchedTokenException e = new MismatchedTokenException('}', input);
        String msg = "missing final '}' in {...} anonymous template";
        group.errMgr.groupLexerError(ErrorType.SYNTAX_ERROR, getSourceName(), e, msg);
        break;
    }
    t = lexer.nextToken();
}
// don't match '}' here; our little {...} scanner loop matches it
// to terminate.
;
COMMENT
   : '/\*' ( options {greedy=false;} : .)* '\*/' {skip();}
   ;

LINE_COMMENT
   : '// '~('
'|'')* ''? '
' {skip();}
   ;

WS :	(' '|''|'	'|'
') {skip();} 

Found in path(s):
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/compiler/Group.g
No license file was found, but licenses were detected in source scan.

/*
 * [The "BSD license"]
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** Build an AST from a single StringTemplate template */
parser grammar STParser;

options {
    language=Java;
    tokenVocab=STLexer;
    TokenLabelType=CommonToken;
    output=AST;
    ASTLabelType=CommonTree;
}
tokens {
    EXPR; OPTIONS; PROP; PROP_IND; INCLUDE; INCLUDE_IND; EXEC_FUNC; INCLUDE_SUPER;
    INCLUDE_SUPER_REGION; INCLUDE_REGION; TO_STR; LIST; MAP; ZIP; SUBTEMPLATE; ARGS;
    ELEMENTS; REGION; NULL; INDENTED_EXPR;
}

@header {
    package org.stringtemplate.v4.compiler;
    import org.stringtemplate.v4.misc.ErrorManager;
    import org.stringtemplate.v4.misc.ErrorType;
}

@members {
    ErrorManager errMgr;
Token templateToken;
public STParser(TokenStream input, ErrorManager errMgr, Token templateToken) {
    this(input);
    this.errMgr = errMgr;
    this.templateToken = templateToken;
}

@Override
protected Object recoverFromMismatchedToken(IntStream input, int ttype, BitSet follow)
throws RecognitionException {
    throw new MismatchedTokenException(ttype, input);
}

@rulecatch {
    catch (RecognitionException re) { throw re; }
}

templateAndEOF : template EOF -> template? ;

template : element* ;

element :
    [{input.LT(1).getCharPositionInLine()==0]? INDENT? COMMENT NEWLINE -> // throw away
    |INDENT singleElement -> ^(INDENTED_EXPR INDENT singleElement?) // singleElement is optional to handle
    |singleElement |
    |compoundElement |
    ;

singleElement :
    exprTag
    |
    TEXT
    |
    NEWLINE
    |
    COMMENT! // throw away
    ;

compoundElement :
    ifstat
    |
    region 
    ;

exprTag :
    LDELIM expr ( ';' exprOptions )? RDELIM
    -> ^(EXPR[SLDELIM,"EXPR"] expr exprOptions?)
    ;

region
(p='.prop=ID-> ^(PROP[$p,"PROP"] $notConditionalExpr $prop)
|p='.' '(' mapExpr ')'-^>(PROP_IND[$p,"PROP_IND"] $notConditionalExpr mapExpr)
)*
;

exprOptions : option ( ',' option )* -> ^(OPTIONS option*) ;

option
@init {
String id = input.LT(1).getText();
String defVal = Compiler.defaultOptionValues.get(id);
boolean validOption = Compiler.supportedOptions.get(id)!null;
}

:ID{
if ( !validOption ) {
    errMgr.compileTimeError(ErrorType.NO_SUCH_OPTION, templateToken, $ID, $ID.text);
}
}

(=' exprNoComma -> {validOption}? ^('= ID exprNoComma)
->
|
if ( defVal==null ) {
    errMgr.compileTimeError(ErrorType.NO_DEFAULT_VALUE, templateToken, $ID);
}
-> {validOption&&defVal!=null}? (^EQUALS[=""] ID STRING[$ID,'"'+defVal+'"'])
->
)
;

exprNoComma
:memberExpr

(.' mapTemplateRef-> ^(MAP memberExpr mapTemplateRef)
|-> memberExpr
)
;

expr : mapExpr :

// more complicated than necessary to avoid backtracking, which ruins
// error handling

mapExpr
:memberExpr

( (c='.' memberExpr)+ col='.' mapTemplateRef
-> ^(ZIP[$col] ^(ELEMENTS memberExpr+) mapTemplateRef)
|-> memberExpr
)
(if ($x!=null) $x.clear();) // don't keep queueing x; new list for each iteration
col='.' x+=mapTemplateRef ($c==null)=> ', x+=mapTemplateRef )*
-> ^(MAP[Scol] $mapExpr $x+)
)*
;

/**
 expr:template(args) apply template to expr
 expr:[arg | ...] apply subtemplate to expr
 expr:(e)(args) convert e to a string template name and apply to expr
 */
mapTemplateRef
:qualifiedId (' args ')-> ^(INCLUDE qualifiedId args?)
| subtemplate
| lp='(' mapExpr rp=')' (' argExprList? ')') -> ^(INCLUDE_IND mapExpr argExprList?)
| ;

memberExpr
:(includeExpr->includeExpr)
(p='.' ID-> ^(PROP[$p,"PROP"] $memberExpr ID)
| p='.' (' mapExpr ')') -> ^(PROP_IND[$p,"PROP_IND"] $memberExpr mapExpr)
)*
;

includeExpr
:options {k=2;} // prevent full LL(*), which fails, falling back on k=1; need k=2
| {Compiler.funcs.containsKey(input.LT(1).getText())}? // predefined function
| ID (' expr? ') -> ^(EXEC_FUNC ID expr?)
| 'super' '.' ID (' args ') -> ^(INCLUDE_SUPER ID args?)
| qualifiedId (' args ') -> ^(INCLUDE qualifiedId args?)
| '@' 'super' '.' ID rp=')' -> ^(INCLUDE_SUPER_REGION ID)
| '@' ID rp=')' -> ^(INCLUDE_REGION ID)
| primary
;

primary
:ID
| STRING
| TRUE
| FALSE
| subtemplate
| list
| [{conditional.size()>0}?=> ('! conditional ')]
| [{conditional.size()==0}?=> lp=(' expr ')'
| (' argExprList? ') -> ^(INCLUDE_INDD[lp] expr argExprList?)
| -> ^(TO_STR[lp] expr)
)
qualifiedId
: (ID-> ID | r ID-> ^(/ r ID) )
( r r=ID-> ^(/ r $qualifiedId $r ) )* ;

args:argExprList
| namedArg ( ',' namedArg )* (',' '...')? -> namedArg+ '...'? |
| '...' |
|
;

argExprList : arg ( ',' arg )* -> arg+ ;

arg : exprNoComma ;

namedArg : ID '=' arg -> ^('=' ID arg) ;

list:{input.LA(2)==RBRACK}? // hush warning: [] special case
| lb='[' -> LIST[$lb]
| [lb='[' listElement ( ',' listElement )* ']' -> ^(LIST[$lb] listElement*) | ;

listElement : exprNoComma | -> NULL ;

Found in path(s):
* /opt/cola/permits/1258286456_1643052484.48/0/st4-4-1-sources-jar/org/stringtemplate/v4/compiler/STParser.g

1.639 ee-first 1.1.1 1.1.1
1.639.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.640 path-to-regexp 1.8.0 1.8.0

1.640.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.641 mime-lookup 0.0.2 0.0.2

1.641.1 Available under license:
Original work Copyright (c) 2010 Benjamin Thomas, Robert Kieffer
Modified work Copyright 2015 Jayson Harshbarger

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.642 globals 9.18.0

1.642.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.643 iconv-lite 0.4.19 0.4.19

1.643.1 Available under license:
Copyright (c) 2011 Alexander Shtuchkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.644 dashify 0.2.2 0.2.2

1.644.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.645 babel-helper-hoist-variables 6.24.1
1.645.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA

CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.646 ipaddr.js 1.8.0 1.8.0

1.646.1 Available under license:

Copyright (C) 2011-2017 whitequark <whitequark@whitequark.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.647 fs-constants 1.0.0 1.0.0

1.647.1 Available under license:

The MIT License (MIT)

Copyright (c) 2018 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
1.648 isbinaryfile 3.0.3 3.0.3
1.648.1 Available under license:
Copyright (c) 2017 Garen J. Torikian

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.649 @babel/plugin-transform-named-capturing-groups-regex 7.16.0
1.649.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.650 babylon 6.18.0
1.650.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.651 swagger-annotations 1.5.16
1.651.1 Available under license:
/**
 * Copyright 2016 SmartBear Software
 * <p>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p>
* http://www.apache.org/licenses/LICENSE-2.0
* <p>
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

package io.swagger.annotations;

import java.lang.annotation.ElementType;
import java.lang.annotation.Retention;
import java.lang.annotation.RetentionPolicy;
import java.lang.annotation.Target;

/**
 * License metadata available within the info section of a Swagger definition, see
 * https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md#licenseObject
 *
 * @since 1.5.0
 */

@Target(ElementType.ANNOTATION_TYPE)
@Retention(RetentionPolicy.RUNTIME)
public @interface License {

/**
 * The name of the license.
 *
 * @return the name of the license
 */
String name();

/**
 * An optional URL for the license.
 *
 * @return an optional URL for the license.
 */
String url() default "";
}
1.652.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.
1.653 hk2-utils 2.2.0-b21

1.653.1 Available under license:

Found license 'General Public License 2.0' in * Copyright (c) 2007-2011 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'. Found license 'Common Development and Distribution License (CDDL) 1.1' in Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'.

Found license 'General Public License 2.0' in * Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'.

Found license 'General Public License 2.0' in Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'.

Found license 'General Public License 2.0' in * Copyright (c) 2012-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'.

Found license 'General Public License 2.0' in * Copyright (c) 2012 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'.

1.654 swagger 2.0.7

1.654.1 Available under license:

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.655 protobuf-java 3.14.0

1.655.1 Available under license:

No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/LazyStringList.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/RpcCallback.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/google/protobuf/wrappers.proto
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/RepeatedFieldBuilderV3.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/Message.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/RawMessageInfo.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/ProtocolStringList.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/Utf8.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/google/protobuf/type.proto
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/CodedOutputStream.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/BufferAllocator.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/ServiceException.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/MessageInfoFactory.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/ServiceException.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
jar/com/google/protobuf/MessageInfoFactory.java
* /opt/cola/permits/1127748132_1611862246.14/0/protobuf-java-3-14-0-sources-
Open Source Used In DNAC Platform Arica 2413
1.656 pend 1.2.0 1.2.0

1.656.1 Available under license:

The MIT License (Expat)

Copyright (c) 2014 Andrew Kelley

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.657 llvm 6.0-1ubuntu2~16.04.1

1.657.1 Available under license:

; RUN: llc < %s -mtriple=s390x-linux-gnu -mcpu=zEC12 -verify-machineinstrs | FileCheck %s
; Test that early if conversion produces LOCR with operands of the right ; register classes.

define void @autogen_SD4739(i8*) {
; CHECK-NOT: Expected a GR32Bit register, but got a GRX32Bit register
BB:
  %L34 = load i8, i8* %0
  %Cmp56 = icmp sgt i8 undef, %L34
  br label %CF246

CF246:                                  ; preds = %CF246, %BB
  %SI163 = select i1 %Cmp56, i8 %L34, i8 undef
  br i1 undef, label %CF246, label %CF248
define void @add256(<16 x i32>* %a, <16 x i32>* %b, <16 x i32>* %c) "min-legal-vector-width"="256" {
 ; CHECK-LABEL: add256:
 ; CHECK:       # %bb.0:
 ; CHECK-NEXT: vmovdqa (%rdi), %ymm0
 ; CHECK-NEXT: vmovdqa 32(%rdi), %ymm1
 ; CHECK-NEXT: vpadd (%rsi), %ymm0, %ymm0
 ; CHECK-NEXT: vpadd 32(%rsi), %ymm1, %ymm1
 ; CHECK-NEXT: vmovdqa %ymm1, 32(%rdx)
 ; CHECK-NEXT: vmovdqa %ymm0, (%rdx)
 ; CHECK-NEXT: vzeroupper
 ; CHECK-NEXT: retq
 %d = load <16 x i32>, <16 x i32>* %a
 %e = load <16 x i32>, <16 x i32>* %b
 %f = add <16 x i32> %d, %e
 store <16 x i32> %f, <16 x i32>* %c
 ret void
}

define void @add512(<16 x i32>* %a, <16 x i32>* %b, <16 x i32>* %c) "min-legal-vector-width"="512" {
 ; CHECK-LABEL: add512:
 ; CHECK:       # %bb.0:
 ; CHECK-NEXT: vmovdqa64 (%rdi), %zmm0
 ; CHECK-NEXT: vpadd (%rsi), %zmm0, %zmm0
 ; CHECK-NEXT: vmovdqa64 %zmm0, (%rdx)
 ; CHECK-NEXT: vzeroupper
 ; CHECK-NEXT: retq
 %d = load <16 x i32>, <16 x i32>* %a
 %e = load <16 x i32>, <16 x i32>* %b
 %f = add <16 x i32> %d, %e
 store <16 x i32> %f, <16 x i32>* %c
 ret void
}

define void @avg_v64i8_256(<64 x i8>* %a, <64 x i8>* %b) "min-legal-vector-width"="256" {
 ; CHECK-LABEL: avg_v64i8_256:
 ; CHECK:       # %bb.0:
 ; CHECK-NEXT: vmovdqa (%rsi), %ymm0
store <64 x i8> %8, <64 x i8>* undef, align 4
ret void
}

define void @pmaddwd_32_256(<32 x i16>* %APtr, <32 x i16>* %BPtr, <16 x i32>* %CPtr) "min-legal-vector-width"="256" {
; CHECK-LABEL: pmaddwd_32_256:
; CHECK: # %bb.0:
; CHECK-NEXT: vmovdqa (%rdi), %ymm0
; CHECK-NEXT: vmovdqa 32(%rdi), %ymm1
; CHECK-NEXT: vpmaddwd (%rsi), %ymm0, %ymm0
; CHECK-NEXT: vpmaddwd 32(%rsi), %ymm1, %ymm1
; CHECK-NEXT: vmovdqa %ymm1, 32(%rdx)
; CHECK-NEXT: vmovdqa %ymm0, (%rdx)
; CHECK-NEXT: vzeroupper
; CHECK-NEXT: retq
%A = load <32 x i16>, <32 x i16>* %APtr
%B = load <32 x i16>, <32 x i16>* %BPtr
%a = sext <32 x i16> %A to <32 x i32>
%b = sext <32 x i16> %B to <32 x i32>
%m = mul nsw <32 x i32> %a, %b
%odd = shufflevector <32 x i32> %m, <32 x i32> undef, <16 x i32> <i32 0, i32 2, i32 4, i32 6, i32 8, i32 10, i32 12, i32 14, i32 16, i32 18, i32 20, i32 22, i32 24, i32 26, i32 28, i32 30>
%even = shufflevector <32 x i32> %m, <32 x i32> undef, <16 x i32> <i32 1, i32 3, i32 5, i32 7, i32 9, i32 11, i32 13, i32 15, i32 17, i32 19, i32 21, i32 23, i32 25, i32 27, i32 29, i32 31>
%ret = add <16 x i32> %odd, %even
store <16 x i32> %ret, <16 x i32>* %CPtr
ret void
}

define void @pmaddwd_32_512(<32 x i16>* %APtr, <32 x i16>* %BPtr, <16 x i32>* %CPtr) "min-legal-vector-width"="512" {
; CHECK-LABEL: pmaddwd_32_512:
; CHECK: # %bb.0:
; CHECK-NEXT: vmovdqa64 (%rdi), %zmm0
; CHECK-NEXT: vpmaddwd (%rsi), %zmm0, %zmm0
; CHECK-NEXT: vmovdqa64 %zmm0, (%rdx)
; CHECK-NEXT: vzeroupper
; CHECK-NEXT: retq
%A = load <32 x i16>, <32 x i16>* %APtr
%B = load <32 x i16>, <32 x i16>* %BPtr
%a = sext <32 x i16> %A to <32 x i32>
%b = sext <32 x i16> %B to <32 x i32>
%m = mul nsw <32 x i32> %a, %b
%odd = shufflevector <32 x i32> %m, <32 x i32> undef, <16 x i32> <i32 0, i32 2, i32 4, i32 6, i32 8, i32 10, i32 12, i32 14, i32 16, i32 18, i32 20, i32 22, i32 24, i32 26, i32 28, i32 30>
%even = shufflevector <32 x i32> %m, <32 x i32> undef, <16 x i32> <i32 1, i32 3, i32 5, i32 7, i32 9, i32 11, i32 13, i32 15, i32 17, i32 19, i32 21, i32 23, i32 25, i32 27, i32 29, i32 31>
%ret = add <16 x i32> %odd, %even
store <16 x i32> %ret, <16 x i32>* %CPtr
ret void
}

define void @psubus_64i8_max_256(<64 x i8>* %xptr, <64 x i8>* %yptr, <64 x i8>* %zptr) "min-legal-vector-width"="256" {
; CHECK-LABEL: psubus_64i8_max_256:
; CHECK:
; # %bb.0:
; CHECK-NEXT: vmovdqa (%rdi), %ymm0
; CHECK-NEXT: vmovdqa 32(%rdi), %ymm1
; CHECK-NEXT: vpsubusb (%rsi), %ymm0, %ymm0
; CHECK-NEXT: vpsubusb 32(%rsi), %ymm1, %ymm1
; CHECK-NEXT: vmovdqa %ymm1, 32(%rdx)
; CHECK-NEXT: vmovdqa %ymm0, (%rdx)
; CHECK-NEXT: vzeroupper
; CHECK-NEXT: retq
%x = load <64 x i8>, <64 x i8>* %xptr
%y = load <64 x i8>, <64 x i8>* %yptr
%cmp = icmp ult <64 x i8> %x, %y
%max = select <64 x i1> %cmp, <64 x i8> %y, <64 x i8> %x
%res = sub <64 x i8> %max, %y
store <64 x i8> %res, <64 x i8>* %zptr
ret void
}

define void @psubus_64i8_max_512(<64 x i8>* %xptr, <64 x i8>* %yptr, <64 x i8>* %zptr) "min-legal-vector-width"="512" {
; CHECK-LABEL: psubus_64i8_max_512:
; CHECK:
; # %bb.0:
; CHECK-NEXT: vmovdqa64 (%rdi), %zmm0
; CHECK-NEXT: vpsubusb (%rsi), %zmm0, %zmm0
; CHECK-NEXT: vmovdqa64 %zmm0, (%rdx)
; CHECK-NEXT: vzeroupper
; CHECK-NEXT: retq
%x = load <64 x i8>, <64 x i8>* %xptr
%y = load <64 x i8>, <64 x i8>* %yptr
%cmp = icmp ult <64 x i8> %x, %y
%max = select <64 x i1> %cmp, <64 x i8> %y, <64 x i8> %x
%res = sub <64 x i8> %max, %y
store <64 x i8> %res, <64 x i8>* %zptr
ret void
}

define i32 @_Z9test_charPcS_i_256(i8* nocapture readonly, i8* nocapture readonly, i32) "min-legal-vector-width"="256" {
; CHECK-LABEL: _Z9test_charPcS_i_256:
; CHECK:
; # %bb.0: # %entry
movl %edx, %eax
vpxor %xmm0, %xmm0, %xmm0
xorl %ecx, %ecx
vpxor %xmm1, %xmm1, %xmm1
vpxor %xmm2, %xmm2, %xmm2
.p2align 4, 0x90
.LBB8_1: # %vector.body
# =>This Inner Loop Header: Depth=1
vpmovsxbw (%rdi,%rcx), %ymm3
vpmovsxbw 16(%rdi,%rcx), %ymm4
vpmovsxbw (%rsi,%rcx), %ymm5
vpmaddwd %ymm3, %ymm5, %ymm3
vpmaddwd %ymm1, %ymm3, %ymm1
vpmaddwd %ymm2, %ymm3, %ymm2
vpmaddwd %ymm0, %ymm1, %ymm0
vpmaddwd %ymm0, %ymm2, %ymm0
addq $32, %rcx
cmpq %rcx, %rax
jne .LBB8_1
vpaddd %ymm0, %ymm1, %ymm1
vpaddd %ymm0, %ymm2, %ymm0
vpaddd %ymm0, %ymm1, %ymm0
vextracti128 $1, %ymm0, %xmm1
vpshufd {%.*#+} xmm1 = xmm0[2,3,0,1]
vpaddd %ymm1, %ymm0, %ymm0
vpshufd {%.*#+} xmm1 = xmm0[1,1,2,3]
vpaddd %xmm1, %xmm0, %eax
vzeroupper
retq
eval:
%3 = zext i32 %2 to i64
br label %vector.body

vector.body:
%index = phi i64 [ %index.next, %vector.body ], [ 0, %entry ]
%vec.phi = phi <32 x i32> [ %11, %vector.body ], [ zeroinitializer, %entry ]
%4 = getelementptr inbounds i8, i8* %0, i64 %index
%5 = bitcast i8* %4 to <32 x i8>*
%wide.load = load <32 x i8>, <32 x i8>* %5, align 1
%6 = sext <32 x i8> %wide.load to <32 x i32>
%7 = getelementptr inbounds i8, i8* %1, i64 %index
%8 = bitcast i8* %7 to <32 x i8>*
%wide.load14 = load <32 x i8>, <32 x i8>* %8, align 1
%9 = sext <32 x i8> %wide.load14 to <32 x i32>
%10 = mul nsw <32 x i32> %9, %6
%11 = add nsw <32 x i32> %10, %vec.phi
%index.next = add i64 %index, 32
%12 = icmp eq i64 %index.next, %3
br i1 %12, label %middle.block, label %vector.body

middle.block:
%rdx.shuf1 = shufflevector <32 x i32> %11, <32 x i32> undef, <32 x i32> <i32 16, i32 17, i32 18, i32 19, i32 20, i32 21, i32 22, i32 23, i32 24, i32 25, i32 26, i32 27, i32 28, i32 29, i32 30, i32 31, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 unde

%13 = extractelement <32 x i32> %bin.rdx20, i32 0
ret i32 %13
}

define i32 @_Z9test_charPcS_i_512(i8* nocapture readonly, i8* nocapture readonly, i32) "min-legal-vector-width"="512" {
; CHECK-LABEL: _Z9test_charPcS_i_512:
; CHECK: # %bb.0: # %entry
; CHECK-NEXT: movl %edx, %eax
; CHECK-NEXT: vpxor %xmm0, %xmm0, %xmm0
; CHECK-NEXT: xorl %ecx, %ecx
; CHECK-NEXT: vpxor %xmm1, %xmm1, %xmm1
; CHECK-NEXT: .p2align 4, 0x90
; CHECK-NEXT: .LBB9_1: # %vector.body
; CHECK-NEXT: # =>This Inner Loop Header: Depth=1
; CHECK-NEXT: vpmovsxbw (%rdi,%rcx), %zmm2
; CHECK-NEXT: vpmovsxbw (%rsi,%rcx), %zmm3
; CHECK-NEXT:  vpmaddwd %zmm2, %zmm3, %zmm2
; CHECK-NEXT:  vpaddd %zmm1, %zmm2, %zmm1
; CHECK-NEXT:  addq $32, %rcx
; CHECK-NEXT:  cmpq %rcx, %rax
; CHECK-NEXT:  jne .LBB9_1
; CHECK-NEXT:  # %bb.2: # %middle.block
; CHECK-NEXT:  vpaddd %zmm0, %zmm1, %zmm0
; CHECK-NEXT:  vextracti64x4 %1, %zmm0, %ymm1
; CHECK-NEXT:  vpaddd %zmm1, %zmm0, %zmm0
; CHECK-NEXT:  vextracti128 %1, %ymm0, %xmm1
; CHECK-NEXT:  vpaddd %zmm1, %zmm0, %zmm0
; CHECK-NEXT:  vpsuhf [%.*++] xmm1 = xmm0[2,3,0,1]
; CHECK-NEXT:  vpaddd %zmm1, %zmm0, %zmm0
; CHECK-NEXT:  vpshufd [%.*++] xmm1 = xmm0[1,2,3]
; CHECK-NEXT:  vpaddd %zmm1, %zmm0, %zmm0
; CHECK-NEXT:  vmovd %xmm1, %eax
; CHECK-NEXT:  vzeroupper
; CHECK-NEXT:  retq

entry:
  %3 = zext i32 %2 to i64
br label %vector.body

vector.body:
  %index = phi i64 [ %index.next, %vector.body ], [ 0, %entry ]
  %vec.phi = phi <32 x i32> [ %11, %vector.body ], [ zeroinitializer, %entry ]
  %4 = getelementptr inbounds i8, i8* %0, i64 %index
  %5 = bitcast i8* %4 to <32 x i8>*
  %wide.load = load <32 x i8>, <32 x i8>* %5, align 1
  %6 = sext <32 x i8> %wide.load to <32 x i32>
  %7 = getelementptr inbounds i8, i8* %1, i64 %index
  %8 = bitcast i8* %7 to <32 x i8>*
  %wide.load14 = load <32 x i8>, <32 x i8>* %8, align 1
  %9 = sext <32 x i8> %wide.load14 to <32 x i32>
  %10 = mul nsw <32 x i32> %9, %6
  %11 = add nsw <32 x i32> %10, %vec.phi
  %index.next = add i64 %index, 32
  %12 = icmp eq i64 %index, %3
br i1 %12, label %middle.block, label %vector.body

middle.block:
  %rdx.shuf1 = shufflevector <32 x i32> %11, %32 x i32> undef, <32 x i32> <i32 16, i32 17, i32 18, i32 19, i32 20, i32 21, i32 22, i32 23, i32 24, i32 25, i32 26, i32 27, i32 28, i32 29, i32 30, i32 31, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 unde
i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef

%bin.rdx = add <32 x i32> %r dx, %rdx.shuf
%rdx.shuf15 = shufflevector <32 x i32> bin.rdx, <32 x i32> undef, <32 x i32> <i32 4, i32 5, i32 6, i32 7, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 unde
%bin.rdx32 = add <32 x i32> %bin.rdx, %rdx.shuf15
%rdx.shuf17 = shufflevector <32 x i32> %bin.rdx32, <32 x i32> undef, <32 x i32> <i32 2, i32 3, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 unde
%bin.rdx18 = add <32 x i32> %bin.rdx32, %rdx.shuf17
%rdx.shuf19 = shufflevector <32 x i32> %bin.rdx18, <32 x i32> undef, <32 x i32> <i32 1, i32 undef, i32 undef, i32 unde
%bin.rdx20 = add <32 x i32> %bin.rdx18, %rdx.shuf19
%13 = extractelement <32 x i32> %bin.rdx20, i32 0
ret i32 %13

define i32 @sad_16i8_256() "min-legal-vector-width"="256" {
; CHECK-LABEL: sad_16i8_256:
; CHECK: % #bb.0: # %entry
; CHECK-NEXT: vpxor %xmm0, %xmm0, %xmm0
; CHECK-NEXT: movq $-1024, %rax # imm = 0xFC00
; CHECK-NEXT: vpxor %xmm1, %xmm1, %xmm1
; CHECK-NEXT: .p2align 4, 0x90
; CHECK-NEXT: .LBB10_1: # %vector.body
; CHECK-NEXT: # =>This Inner Loop Header: Depth=1
; CHECK-NEXT: vmovdqu a+1024(%rax), %ymm2
; CHECK-NEXT: vpsadbw b+1024(%rax), %ymm2, %ymm2
; CHECK-NEXT: vpaddd %ymm1, %ymm2, %ymm1
; CHECK-NEXT: addq $4, %rax
; CHECK-NEXT: jne .LBB10_1
; CHECK-NEXT: % #bb.2: # %middle.block
; CHECK-NEXT: vpaddd %ymm0, %ymm1, %ymm0
; CHECK-NEXT: vextracti128 %ymm2, %ymm0, %ymm0
; CHECK-NEXT: vpaddd %ymm1, %ymm0, %ymm0
; CHECK-NEXT: vpsdf [[*#*]] %ymm1 = %ymm0(2,3,0,1]
; CHECK-NEXT: vpaddd %ymm1, %ymm0, %ymm0
; CHECK-NEXT: vpsdf [[*#*]] %ymm1 = %ymm0(1,1,2,3]
; CHECK-NEXT: vpaddd %ymm1, %ymm0, %ymm0
; CHECK-NEXT: vmovd %xmm0, %eax
; CHECK-NEXT: vzercupper

; CHECK-NEXT:   retq
entry:
br label %vector.body

vector.body:

%index = phi i64 [ 0, %entry ], [ %index.next, %vector.body ]
%vec.phi = phi <16 x i32> [ zeroinitializer, %entry ], [ %10, %vector.body ]
%0 = getelementptr inbounds [1024 x i8], [1024 x i8]* @a, i64 0, i64 %index
%1 = bitcast i8* %0 to <16 x i8>*
%wide.load = load <16 x i8>, <16 x i8>* %1, align 4
%2 = zext <16 x i8> %wide.load to <16 x i32>
%3 = getelementptr inbounds [1024 x i8], [1024 x i8]* @b, i64 0, i64 %index
%4 = bitcast i8* %3 to <16 x i8>*
%wide.load1 = load <16 x i8>, <16 x i8>* %4, align 4
%5 = zext <16 x i8> %wide.load1 to <16 x i32>
%6 = sub nsw <16 x i32> %2, %5
%7 = icmp sgt <16 x i32> %6, <i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1>
%8 = sub nsw <16 x i32> zeroinitializer, %6
%9 = select <16 x i1> %7, <16 x i32> %6, <16 x i32> %8
%10 = add nsw <16 x i32> %9, %vec.phi
%index.next = add i64 %index, 4
br i1 %11, label %middle.block, label %vector.body

middle.block:

%.lcssa = phi <16 x i32> [ %10, %vector.body ]
%rdx.shuf = shufflevector <16 x i32> %.lcssa, <16 x i32> undef, <16 x i32> <i32 8, i32 9, i32 10, i32 11, i32 12, i32 13, i32 14, i32 15, i32 undef>, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%bin.rdx = add <16 x i32> %.lcssa, %rdx.shuf
%rdx.shuf2 = shufflevector <16 x i32> %bin.rdx, <16 x i32> undef, <16 x i32> <i32 4, i32 5, i32 6, i32 7, i32 8, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%bin.rdx2 = add <16 x i32> %bin.rdx, %rdx.shuf2
%rdx.shuf3 = shufflevector <16 x i32> %bin.rdx2, <16 x i32> undef, <16 x i32> <i32 2, i32 3, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%bin.rdx3 = add <16 x i32> %bin.rdx2, %rdx.shuf3
%rdx.shuf4 = shufflevector <16 x i32> %bin.rdx3, <16 x i32> undef, <16 x i32> <i32 1, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%bin.rdx4 = add <16 x i32> %bin.rdx3, %rdx.shuf4
%12 = extractelement <16 x i32> %bin.rdx4, i32 0
ret i32 %12
}
define i32 @sad_16i8_512() "min-legal-vector-width"="512" {
; CHECK-LABEL: sad_16i8_512:
Open Source Used In DNAC Platform Arica

vector.body:

%index = phi i64 [ 0, %entry ], [ %index.next, %vector.body ]
%vec.phi = phi <16 x i32> [ zeroinitializer, %entry ], [ %10, %vector.body ]
%0 = getelementptr inbounds [1024 x i8], [1024 x i8]* @a, i64 0, i64 %index
%1 = bitcast i8* %0 to <16 x i8>*
%wide.load = load <16 x i8>, <16 x i8>* %1, align 4
%2 = zext <16 x i8> %wide.load to <16 x i32>
%3 = getelementptr inbounds [1024 x i8], [1024 x i8]* @b, i64 0, i64 %index
%4 = bitcast i8* %3 to <16 x i8>*
%wide.load1 = load <16 x i8>, <16 x i8>* %4, align 4
%5 = zext <16 x i8> %wide.load1 to <16 x i32>
%6 = sub nsw <16 x i32> %2, %5
%7 = icmp sgt <16 x i32> %6, <i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1, i32 -1> %8 = sub nsw <16 x i32> zeroinitializer, %6
%9 = select <16 x i1> %7, <16 x i32> %6, <16 x i32> %8
%10 = add nsw <16 x i32> %9, %vec.phi
%index.next = add i64 %index, 1024
br i1 %11, label %middle.block, label %vector.body

middle.block:
%.lcssa = phi <16 x i32> [ %10, %vector.body ]
%.rdx.shuf = shufflevector <16 x i32> %.lcssa, <16 x i32> undef, <16 x i32> <i32 8, i32 9, i32 10, i32 11, i32 12, i32 13, i32 14, i32 15, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%.bin.rdx = add <16 x i32> %.lcssa, %.rdx.shuf
%.rdx.shuf2 = shufflevector <16 x i32> <16 x i32> undef, <16 x i32> <i32 4, i32 5, i32 6, i32 7, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%.bin.rdx2 = add <16 x i32> %.bin.rdx, %.rdx.shuf2
%.rdx.shuf3 = shufflevector <16 x i32> <16 x i32> undef, <16 x i32> <i32 2, i32 3, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%.bin.rdx3 = add <16 x i32> %.bin.rdx2, %.rdx.shuf3
%.rdx.shuf4 = shufflevector <16 x i32> <16 x i32> undef, <16 x i32> <i32 1, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef, i32 undef>
%.bin.rdx4 = add <16 x i32> %.bin.rdx3, %.rdx.shuf4
%12 = extractelement <16 x i32> %.bin.rdx4, i32 0
ret i32 %12
}
define void @sbto16f32_256(<16 x i16> %a, <16 x float>* %res) "min-legal-vector-width"="256" {
    ; CHECK-LABEL: sbto16f32_256:
    ; CHECK:       # %bb.0:
    ; CHECK-NEXT:  vpmovw2m %ymm0, %k0
    ; CHECK-NEXT:  kshiftrw $8, %k0, %k1
    ; CHECK-NEXT:  vpmovm2d %k1, %ymm0
    ; CHECK-NEXT:  vcvtdq2ps %ymm0, %ymm0
    ; CHECK-NEXT:  vpmovm2d %k0, %ymm1
    ; CHECK-NEXT:  vcvtdq2ps %ymm1, %ymm1
    ; CHECK-NEXT:  vmovaps %ymm1, (%rdi)
    ; CHECK-NEXT:  vmovaps %ymm0, 32(%rdi)
    ; CHECK-NEXT:  vzeroupper
    ; CHECK-NEXT:  retq
    %mask = icmp slt <16 x i16> %a, zeroinitializer
    %1 = sitofp <16 x i1> %mask to <16 x float>
    store <16 x float> %1, <16 x float>* %res
    ret void
}
define void @sbto16f32_512(<16 x i16> %a, <16 x float>* %res) "min-legal-vector-width"="512" {
    ; CHECK-LABEL: sbto16f32_512:
    ; CHECK:       # %bb.0:
    ; CHECK-NEXT:  vpmovw2m %ymm0, %k0
    ; CHECK-NEXT:  vpmovm2d %k1, %ymm0
    ; CHECK-NEXT:  vcvtdq2ps %ymm0, %ymm0
    ; CHECK-NEXT:  vpmovm2d %k0, %ymm1
    ; CHECK-NEXT:  vcvtdq2ps %ymm1, %ymm1
    ; CHECK-NEXT:  vmovaps %ymm1, (%rdi)
    ; CHECK-NEXT:  vmovaps %ymm0, 32(%rdi)
    ; CHECK-NEXT:  vzeroupper
    ; CHECK-NEXT:  retq
%mask = icmp slt <16 x i16> %a, zeroinitializer
%1 = sitofp <16 x i1> %mask to <16 x float>
store <16 x float> %1, <16 x float>* %res
ret void
}

define void @sbto16f64_256(<16 x i16> %a, <16 x double>* %res)  "min-legal-vector-width"="256" {  
; CHECK-LABEL: sbto16f64_256:  
; CHECK:  # %bb.0:  
; CHECK-NEXT:  vpmovw2m %ymm0, %k0  
; CHECK-NEXT:  kshiftrw $8, %k0, %k1  
; CHECK-NEXT:  vpmovm2d %k1, %ymm0  
; CHECK-NEXT:  vcvtdq2pd %xmm0, %ymm1  
; CHECK-NEXT:  vextracti128 $1, %ymm0, %xmm0  
; CHECK-NEXT:  vcvtdq2pd %xmm0, %ymm0  
; CHECK-NEXT:  vpmovm2d %k0, %ymm2  
; CHECK-NEXT:  vcvtdq2pd %xmm2, %ymm3  
; CHECK-NEXT:  vextracti128 $1, %ymm2, %xmm2  
; CHECK-NEXT:  vcvtdq2pd %xmm2, %ymm2  
; CHECK-NEXT:  vmovaps %ymm2, 32(%rdi)  
; CHECK-NEXT:  vmovaps %ymm1, 64(%rdi)  
; CHECK-NEXT:  vmovaps %ymm0, 96(%rdi)  
; CHECK-NEXT:  vzeroupper  
; CHECK-NEXT:  retq  
%mask = icmp slt <16 x i16> %a, zeroinitializer
%1 = sitofp <16 x i1> %mask to <16 x float>
store <16 x float> %1, <16 x float>* %res
ret void
}

define void @sbto16f64_512(<16 x i16> %a, <16 x double>* %res)  "min-legal-vector-width"="512" {  
; CHECK-LABEL: sbto16f64_512:  
; CHECK:  # %bb.0:  
; CHECK-NEXT:  vpmovw2m %ymm0, %k0  
; CHECK-NEXT:  vpmovm2d %k0, %zmm0  
; CHECK-NEXT:  vcvtdq2pd %ymm0, %zmm1  
; CHECK-NEXT:  vextracti64x4 $1, %zmm0, %ymm0  
; CHECK-NEXT:  vcvtdq2pd %ymm0, %zmm0  
; CHECK-NEXT:  vmovaps %zmm0, 64(%rdi)  
; CHECK-NEXT:  vmovaps %zmm1, (%rdi)  
; CHECK-NEXT:  vzeroupper  
; CHECK-NEXT:  retq  
%mask = icmp slt <16 x i16> %a, zeroinitializer
%1 = sitofp <16 x i1> %mask to <16 x float>
store <16 x float> %1, <16 x float>* %res
ret void
}
define void @ubto16f32_256(<16 x i16> %a, <16 x float>* %res) "min-legal-vector-width"="256" {
 ; CHECK-LABEL: ubto16f32_256:
 ; CHECK:     # %bb.0:
 ; CHECK-NEXT: vpmovw2m %ymm0, %k0
 ; CHECK-NEXT: kshiftrw $8, %k0, %k1
 ; CHECK-NEXT: vpmovm2d %k1, %ymm0
 ; CHECK-NEXT: vpsrld $31, %ymm0, %ymm0
 ; CHECK-NEXT: vcvtdq2ps %ymm0, %ymm0
 ; CHECK-NEXT: vpmovm2d %k0, %ymm1
 ; CHECK-NEXT: vpsrld $31, %ymm1, %ymm1
 ; CHECK-NEXT: vcvtdq2ps %ymm1, %ymm1
 ; CHECK-NEXT: vmovaps %ymm1, (%rdi)
 ; CHECK-NEXT: vmovaps %ymm0, 32(%rdi)
 ; CHECK-NEXT: vzeroupper
 ; CHECK-NEXT: retq
 %mask = icmp slt <16 x i16> %a, zeroinitializer
 %1 = uitofp <16 x i1> %mask to <16 x float>
 store <16 x float> %1, <16 x float>* %res
 ret void
}

define void @ubto16f32_512(<16 x i16> %a, <16 x float>* %res) "min-legal-vector-width"="512" {
 ; CHECK-LABEL: ubto16f32_512:
 ; CHECK:     # %bb.0:
 ; CHECK-NEXT: vpmovw2m %ymm0, %k0
 ; CHECK-NEXT: vpmovm2d %k0, %zmm0
 ; CHECK-NEXT: vpsrld $31, %zmm0, %zmm0
 ; CHECK-NEXT: vcvtdq2ps %zmm0, %zmm0
 ; CHECK-NEXT: vmovaps %zmm0, (%rdi)
 ; CHECK-NEXT: vzeroupper
 ; CHECK-NEXT: retq
 %mask = icmp slt <16 x i16> %a, zeroinitializer
 %1 = uitofp <16 x i1> %mask to <16 x float>
 store <16 x float> %1, <16 x float>* %res
 ret void
}

define void @ubto16f64_256(<16 x i16> %a, <16 x double>* %res) "min-legal-vector-width"="256" {
 ; CHECK-LABEL: ubto16f64_256:
 ; CHECK:     # %bb.0:
 ; CHECK-NEXT: vpmovw2m %ymm0, %k0
 ; CHECK-NEXT: kshiftrw $8, %k0, %k1
 ; CHECK-NEXT: vpmovm2d %k1, %ymm0
 ; CHECK-NEXT: vpsrld $31, %ymm0, %ymm0
 ; CHECK-NEXT: vcvtdq2pd %xmm0, %ymm0
 ; CHECK-NEXT: vextracti128 $1, %ymm0, %xmm0
 ; CHECK-NEXT: vcvtdq2pd %xmm0, %ymm0

define void @ubto16f64_512(<16 x i16> %a, <16 x double>* %res) "min-legal-vector-width"="512" {  
; CHECK-LABEL: ubto16f64_512:  
; CHECK:   
; CHECK-NEXT: vpmovw2m %ymm0, %k0  
; CHECK-NEXT: vpmovm2d %k0, %zmm0  
; CHECK-NEXT: vpsrl $31, %zmm0, %zmm0  
; CHECK-NEXT: vcvtdq2pd %zmm0, %zmm1  
; CHECK-NEXT: vextracti64x4 $1, %zmm0, %ymm0  
; CHECK-NEXT: vcvtdq2pd %ymm0, %zmm0  
; CHECK-NEXT: vmovaps %zmm0, 64(%rdi)  
; CHECK-NEXT: vmovaps %zmm1, (%rdi)  
; CHECK-NEXT: vzeroupper  
; CHECK-NEXT: retq  
  
%mask = icmp slt <16 x i16> %a, zeroinitializer  
%1 = uitofp <16 x i1> %mask to <16 x double>  
store <16 x double> %1, <16 x double>* %res  
ret void  }

define <16 x i16> @test_16f32toub_256(<16 x float>* %ptr, <16 x i16> %passthru) "min-legal-vector-width"="256" {  
; CHECK-LABEL: test_16f32toub_256:  
; CHECK:   
; CHECK-NEXT: vcvttps2dq (%rdi), %ymm1  
; CHECK-NEXT: vpslld $31, %ymm1, %ymm1  
; CHECK-NEXT: vpmovd2m %ymm1, %k0  
; CHECK-NEXT: vcvttps2dq 32(%rdi), %ymm1  
; CHECK-NEXT: vpslld $31, %ymm1, %ymm1  
; CHECK-NEXT: vpmovd2m %ymm1, %k1  
; CHECK-NEXT: kunpckbw %k0, %k1, %k1  
; CHECK-NEXT: vmovdq16 %ymm0, %ymm0 [%k1] [z]

Open Source Used In DNAC Platform Arica 2430
define <16 x i16> @test_16f32toub_512(<16 x float>* %ptr, <16 x i16> %passthru) "min-legal-vector-width"="512" {
    ; CHECK-LABEL:  test_16f32toub_512:
    ; CHECK:       # %bb.0:
    ; CHECK-NEXT:  vcvtps2dq (%rdi), %zmm1
    ; CHECK-NEXT:  vpslld $31, %zmm1, %zmm1
    ; CHECK-NEXT:  vpmovd2m %zmm1, %k1
    ; CHECK-NEXT:  vmovdqu16 %ymm0, %ymm0 [%k1] {z}
    ; CHECK-NEXT:  retq
    %a = load <16 x float>, <16 x float>* %ptr
    %mask = fptoui <16 x float> %a to <16 x i1>
    %select = select <16 x i1> %mask, <16 x i16> %passthru, <16 x i16> zeroinitializer
    ret <16 x i16> %select
}

define <16 x i16> @test_16f32tosb_256(<16 x float>* %ptr, <16 x i16> %passthru) "min-legal-vector-width"="256" {
    ; CHECK-LABEL:  test_16f32tosb_256:
    ; CHECK:       # %bb.0:
    ; CHECK-NEXT:  vcvtps2dq (%rdi), %ymm1
    ; CHECK-NEXT:  vpmovd2m %ymm1, %k0
    ; CHECK-NEXT:  vcvttps2dq 32(%rdi), %ymm1
    ; CHECK-NEXT:  vpmovd2m %ymm1, %k1
    ; CHECK-NEXT:  kunpckbw %k0, %k1, %k1
    ; CHECK-NEXT:  vmovdqu16 %ymm0, %ymm0 [%k1] {z}
    ; CHECK-NEXT:  retq
    %a = load <16 x float>, <16 x float>* %ptr
    %mask = fptosi <16 x float> %a to <16 x i1>
    %select = select <16 x i1> %mask, <16 x i16> %passthru, <16 x i16> zeroinitializer
    ret <16 x i16> %select
}

define <16 x i16> @test_16f32tosb_512(<16 x float>* %ptr, <16 x i16> %passthru) "min-legal-vector-width"="512" {
    ; CHECK-LABEL:  test_16f32tosb_512:
    ; CHECK:       # %bb.0:
    ; CHECK-NEXT:  vcvtps2dq (%rdi), %zmm1
    ; CHECK-NEXT:  vpmovd2m %zmm1, %k1
    ; CHECK-NEXT:  vmovdqu16 %ymm0, %ymm0 [%k1] {z}
    ; CHECK-NEXT:  retq
    %a = load <16 x float>, <16 x float>* %ptr
    %mask = fptoui <16 x float> %a to <16 x i1>
    %select = select <16 x i1> %mask, <16 x i16> %passthru, <16 x i16> zeroinitializer
    ret <16 x i16> %select
}
%mask = fptosi <16 x float> %a to <16 x i1>
%select = select <16 x i1> %mask, <16 x i16> %passthru, <16 x i16> zeroinitializer
ret <16 x i16> %select
}

define void @mul256(<64 x i8>* %a, <64 x i8>* %b, <64 x i8>* %c) "min-legal-vector-width"="256" {
; CHECK-LABEL: mul256:
; CHECK: # %bb.0:
; CHECK-NEXT: vmovdqa (%rdi), %ymm0
; CHECK-NEXT: vmovdqa 32(%rdi), %ymm1
; CHECK-NEXT: vmovdqa (%rsi), %ymm2
; CHECK-NEXT: vmovdqa 32(%rsi), %ymm3
; CHECK-NEXT: vpunpckhbw {{.*#+}} ymm4 =
  ymm2[8],ymm0[8],ymm2[9],ymm0[9],ymm2[10],ymm0[10],ymm2[11],ymm0[11],ymm2[12],ymm0[12],ymm2[13]
  ,ymm0[13],ymm2[14],ymm0[14],ymm2[15],ymm0[15],ymm2[24],ymm0[24],ymm2[25],ymm0[25],ymm2[26],ymm
  0[26],ymm2[27],ymm0[27],ymm2[28],ymm0[28],ymm2[29],ymm0[29],ymm2[30],ymm0[30],ymm2[31],ymm0[31]
; CHECK-NEXT: vpunpckhbw {{.*#+}} ymm5 =
  ymm0[8,9,9,10,10,11,11,12,12,13,13,14,14,15,24,24,25,25,26,26,27,27,28,28,29,29,30,30,31,31]
; CHECK-NEXT: vmullw %ymm4, %ymm5, %ymm4
; CHECK-NEXT: vmovdqa {{.*#+}} ymm5 =
; CHECK-NEXT: vpand %ymm4, %ymm4, %ymm4
; CHECK-NEXT: vpunpcklbw {{.*#+}} ymm2 =
  ymm2[0],ymm0[0],ymm2[1],ymm0[1],ymm2[2],ymm0[2],ymm2[3],ymm0[3],ymm2[4],ymm0[4],ymm2[5],ymm0[5]
  ,ymm2[6],ymm0[6],ymm2[7],ymm0[7],ymm2[16],ymm0[16],ymm2[17],ymm0[17],ymm2[18],ymm0[18],ymm2[19]
  ,ymm0[19],ymm2[20],ymm0[20],ymm2[21],ymm0[21],ymm2[22],ymm0[22],ymm2[23],ymm0[23]
; CHECK-NEXT: vpunpcklbw {{.*#+}} ymm0 =
  ymm0[0,0,1,1,2,2,3,3,4,4,5,5,6,6,7,7,16,16,17,17,18,18,19,19,20,20,21,21,22,22,23,23]
; CHECK-NEXT: vmullw %ymm2, %ymm0, %ymm0
; CHECK-NEXT: vpadd %ymm5, %ymm0, %ymm0
; CHECK-NEXT: vpunpckuswb {{.*#+}} ymm4 =
  ymm3[8],ymm0[8],ymm3[9],ymm0[9],ymm3[10],ymm0[10],ymm3[11],ymm0[11],ymm3[12],ymm0[12],ymm3[13]
  ,ymm0[13],ymm3[14],ymm0[14],ymm3[15],ymm0[15],ymm3[24],ymm0[24],ymm3[25],ymm0[25],ymm3[26],ymm
  0[26],ymm3[27],ymm0[27],ymm3[28],ymm0[28],ymm3[29],ymm0[29],ymm3[30],ymm0[30],ymm3[31],ymm0[31]
; CHECK-NEXT: vpunpckhbw {{.*#+}} ymm4 =
  ymm1[8],ymm0[8],ymm1[9],ymm0[9],ymm1[10],ymm0[10],ymm1[11],ymm0[11],ymm1[12],ymm0[12],ymm1[13]
  ,ymm0[13],ymm1[14],ymm0[14],ymm1[15],ymm0[15],ymm1[24],ymm0[24],ymm1[25],ymm0[25],ymm1[26],ymm
  0[26],ymm1[27],ymm0[27],ymm1[28],ymm0[28],ymm1[29],ymm0[29],ymm1[30],ymm0[30],ymm1[31],ymm0[31]
; CHECK-NEXT: vmullw %ymm2, %ymm4, %ymm0
; CHECK-NEXT: vpadd %ymm5, %ymm2, %ymm2
; CHECK-NEXT: vpunpckhbw {{.*#+}} ymm3 =
  ymm3[0],ymm0[0],ymm3[1],ymm0[1],ymm3[2],ymm0[2],ymm3[3],ymm0[3],ymm3[4],ymm0[4],ymm3[5],ymm0[5]
  ,ymm3[6],ymm0[6],ymm3[7],ymm0[7],ymm3[16],ymm0[16],ymm3[17],ymm0[17],ymm3[18],ymm0[18],ymm3[19]
  ,ymm0[19],ymm3[20],ymm0[20],ymm3[21],ymm0[21],ymm3[22],ymm0[22],ymm3[23],ymm0[23]
Open Source Used In DNAC Platform Arica
Copyright (c) 2006 Kirill Simonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====================
LLVM Release License
====================

University of Illinois/NCSA
Open Source License

Copyright (c) 2007-2018 University of Illinois at Urbana-Champaign.
All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

http://llvm.org
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal with the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimers.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimers in the documentation and/or other materials provided with the distribution.

* Neither the names of the LLVM Team, University of Illinois at Urbana-Champaign, nor the names of its
contributors may be used to endorse or promote products derived from this Software without specific prior written
permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

The LLVM software contains code written by third parties. Such software will have its own individual
LICENSE.TXT file in the directory in which it appears. This file will describe the copyrights, license, and
restrictions which apply to that code.

The disclaimer of warranty in the University of Illinois Open Source License applies to all code in the LLVM
Distribution, and nothing in any of the other licenses gives permission to use the names of the LLVM Team or the
University of Illinois to endorse or promote products derived from this Software.

The following pieces of software have additional or alternate copyrights, licenses, and/or restrictions:

<table>
<thead>
<tr>
<th>Program</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;none yet&gt;</td>
<td>---------</td>
</tr>
</tbody>
</table>

define void @zext256() "min-legal-vector-width"="256" {
define void @zext512() "min-legal-vector-width"="512" {
    ; AVX-LABEL: 'zext512'
    ; AVX-NEXT: Cost Model: Found an estimated cost of 7 for instruction: %A = zext <8 x i16> undef to <8 x i64>
    ; AVX-NEXT: Cost Model: Found an estimated cost of 3 for instruction: %B = zext <8 x i32> undef to <8 x i64>
    ; AVX-NEXT: Cost Model: Found an estimated cost of 4 for instruction: %C = zext <16 x i8> undef to <16 x i32>
    ; AVX-NEXT: Cost Model: Found an estimated cost of 4 for instruction: %D = zext <16 x i16> undef to <16 x i32>
    ; AVX-NEXT: Cost Model: Found an estimated cost of 3 for instruction: %E = zext <32 x i8> undef to <32 x i16>
    ; AVX-NEXT: Cost Model: Found an estimated cost of 0 for instruction: ret void
    
    ; SKX256-LABEL: 'zext512'
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %A = zext <8 x i16> undef to <8 x i64>
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %B = zext <8 x i32> undef to <8 x i64>
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %C = zext <16 x i8> undef to <16 x i32>
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %D = zext <16 x i16> undef to <16 x i32>
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %E = zext <32 x i8> undef to <32 x i16>
    ; SKX256-NEXT: Cost Model: Found an estimated cost of 0 for instruction: ret void
}
define void @sext256() "min-legal-vector-width"="256" { 
  ; VEC256-LABEL: 'sext256'
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 7 for instruction: %A = sext <8 x i8> undef to <8 x i64>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 7 for instruction: %B = sext <8 x i16> undef to <8 x i64>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 3 for instruction: %C = sext <8 x i32> undef to <8 x i64>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 4 for instruction: %D = sext <16 x i8> undef to <16 x i32>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 4 for instruction: %E = sext <16 x i16> undef to <16 x i32>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 3 for instruction: %F = sext <32 x i8> undef to <32 x i16>
  ; VEC256-NEXT: Cost Model: Found an estimated cost of 0 for instruction: ret void 
  ;
  ; VEC512-LABEL: 'sext512'
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 24 for instruction: %A = sext <8 x i8> undef to <8 x i64>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 24 for instruction: %B = sext <8 x i16> undef to <8 x i64>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 8 for instruction: %C = sext <8 x i32> undef to <8 x i64>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 8 for instruction: %D = sext <16 x i8> undef to <16 x i32>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 8 for instruction: %E = sext <16 x i16> undef to <16 x i32>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 4 for instruction: %F = sext <32 x i8> undef to <32 x i16>
  ; VEC512-NEXT: Cost Model: Found an estimated cost of 0 for instruction: ret void 
  ;
};
define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}

define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}

define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}

define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}

define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}

define void @sext512() "min-legal-vector-width"="512" {
  %A = sext <8 x i8> undef to <8 x i64>
  %B = sext <8 x i16> undef to <8 x i64>
  %C = sext <8 x i32> undef to <8 x i64>
  %D = sext <16 x i8> undef to <16 x i32>
  %E = sext <16 x i16> undef to <16 x i32>
  %F = sext <32 x i8> undef to <32 x i16>
  ret void
}
; VEC512-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %B = sext <8 x i16> undef to <8 x i64>
; VEC512-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %C = sext <8 x i32> undef to <8 x i64>
; VEC512-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %D = sext <16 x i8> undef to <16 x i32>
; VEC512-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %E = sext <16 x i16> undef to <16 x i32>
; VEC512-NEXT: Cost Model: Found an estimated cost of 1 for instruction: %F = sext <32 x i8> undef to <32 x i16>
; VEC512-NEXT: Cost Model: Found an estimated cost of 0 for instruction: ret void
;
%A = sext <8 x i8> undef to <8 x i64>
%B = sext <8 x i16> undef to <8 x i64>
%C = sext <8 x i32> undef to <8 x i64>
%D = sext <16 x i8> undef to <16 x i32>
%E = sext <16 x i16> undef to <16 x i32>
%F = sext <32 x i8> undef to <32 x i16>
ret void
}

$OpenBSD: COPYRIGHT,v 1.3 2003/06/02 20:18:36 millert Exp$

Copyright 1992, 1993, 1994 Henry Spencer. All rights reserved.
This software is not subject to any license of the American Telephone
and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on
any computer system, and to alter it and redistribute it, subject
to the following restrictions:

1. The author is not responsible for the consequences of use of this
   software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by
   explicit claim or by omission. Since few users ever read sources,
   credits must appear in the documentation.

3. Altered versions must be plainly marked as such, and must not be
   misrepresented as being the original software. Since few users
   ever read sources, credits must appear in the documentation.

4. This notice may not be removed or altered.

=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
/*.
 * Copyright (c) 1994
 * The Regents of the University of California. All rights reserved.

Open Source Used In DNAC Platform Arica 2439
Open Source Used In DNAC Platform Arica 2440

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@(#)COPYRIGHT 8.1 (Berkeley) 3/16/94

@(#)COPYRIGHT 8.1 (Berkeley) 3/16/94

; RUN: llc -march=hexagon < %s
; REQUIRES: asserts

; The two loads based on %struct.0, loading two different data types cause LSR to assume type "void" for the memory type. This would then cause an assert in isLegalAddressingMode. Make sure we no longer crash.

target triple = "hexagon"

%struct.0 = type { i8*, i8, %union.anon.0 }
%union.anon.0 = type { i8* }

define hidden fastcc void @fred() unnamed addr #0 {
  entry:
  br i1 undef, label %while.end, label %while.body.lr.ph

  while.body.lr.ph: ; preds = %entry
  br label %while.body

  while.body: ; preds = %exit.2, %while.body.lr.ph
  %lsr.iv = phi %struct.0* [%cgep22, %exit.2 ], [ undef, %while.body.lr.ph ]

...
switch i32 undef, label %exit [  
i32 1, label %sw.bb.i  
i32 2, label %sw.bb3.i  ]

sw.bb.i: ; preds = %while.body  
unreachable

sw.bb3.i: ; preds = %while.body  
unreachable

exit: ; preds = %while.body  
switch i32 undef, label %exit.2 [  
i32 1, label %sw.bb.i17  
i32 2, label %sw.bb3.i20  ]

sw.bb.i17: ; preds = %exit  
%0 = bitcast %struct.0* %lsr.iv to i32*  
%1 = load i32, i32* %0, align 4  
unreachable

sw.bb3.i20: ; preds = %exit  
%2 = bitcast %struct.0* %lsr.iv to i8**  
%3 = load i8*, i8** %2, align 4  
unreachable

exit.2: ; preds = %exit  
%cgep22 = getelementptr %struct.0, %struct.0* %lsr.iv, i32 1  
br label %while.body

while.end: ; preds = %entry  
ret void
}

attributes #0 = { nounwind optsize "target-cpu"="hexagonv55" } ; RUN: llc -mtriple=aarch64-apple-ios %s -o - | FileCheck %s

define <16 x double> @test_sitofp_fixed(<16 x i32> %in) {  
; CHECK-LABEL: test_sitofp_fixed:
  ; First, extend each i32 to i64  
  ; CHECK-DAG: sshll2.2d [[BLOCK0_HI:v[0-9]+]], v0, #0  
  ; CHECK-DAG: sshll2.2d [[BLOCK1_HI:v[0-9]+]], v1, #0  
  ; CHECK-DAG: sshll2.2d [[BLOCK2_HI:v[0-9]+]], v2, #0  
  ; CHECK-DAG: sshll2.2d [[BLOCK3_HI:v[0-9]+]], v3, #0  
  ; CHECK-DAG: sshll2.2d [[BLOCK0_LO:v[0-9]+]], v0, #0  
  ; CHECK-DAG: sshll2.2d [[BLOCK1_LO:v[0-9]+]], v1, #0
; Next, convert each to double.
; CHECK-DAG: scvtf.2d v0, [[[BLOCK0_LO]]]
; CHECK-DAG: scvtf.2d v1, [[[BLOCK0_HI]]]
; CHECK-DAG: scvtf.2d v2, [[[BLOCK1_LO]]]
; CHECK-DAG: scvtf.2d v3, [[[BLOCK1_HI]]]
; CHECK-DAG: scvtf.2d v4, [[[BLOCK2_LO]]]
; CHECK-DAG: scvtf.2d v5, [[[BLOCK2_HI]]]
; CHECK-DAG: scvtf.2d v6, [[[BLOCK3_LO]]]
; CHECK-DAG: scvtf.2d v7, [[[BLOCK3_HI]]]

; CHECK: ret
%flt = sitofp <16 x i32> %in to <16 x double>
%res = fdiv <16 x double> %flt, <double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0>
ret <16 x double> %res }

; This one is small enough to satisfy isSimple, but still illegally large.
define <4 x double> @test_sitofp_fixed_shortish(<4 x i64> %in) {
; CHECK-LABEL: test_sitofp_fixed_shortish:

; CHECK-DAG: scvtf.2d v0, v0
; CHECK-DAG: scvtf.2d v1, v1

; CHECK: ret
%flt = sitofp <4 x i64> %in to <4 x double>
%res = fdiv <4 x double> %flt, <double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0, double 64.0>
ret <4 x double> %res }

; RUN: opt %s -inline -S | FileCheck %s

define internal void @innerSmall() "min-legal-vector-width"="128" {
ret void
}

define internal void @innerLarge() "min-legal-vector-width"="512" {
ret void
}

define internal void @innerNoAttribute() {
ret void
}

; We should not add an attribute during inlining. No attribute means unknown.
Inlining doesn't change the fact that we don't know anything about this
function.

```
define void @outerNoAttribute() {
call void @innerLarge()
ret void
}
```

```
define void @outerConflictingAttributeSmall() "min-legal-vector-width"="128" {
call void @innerLarge()
ret void
}
```

```
define void @outerConflictingAttributeLarge() "min-legal-vector-width"="512" {
call void @innerSmall()
ret void
}
```

We should remove the attribute after inlining since the callee's
vector width requirements are unknown.

```
define void @outerAttribute() "min-legal-vector-width"="128" {
call void @innerNoAttribute()
ret void
}
```

; We should remove the attribute after inlining since the callee's
; vector width requirements are unknown.
define void @outerAttribute() "min-legal-vector-width"="128" {
call void @innerNoAttribute()
ret void
}

; CHECK: define void @outerNoAttribute() {
; CHECK: define void @outerConflictingAttributeSmall() #0
; CHECK: define void @outerConflictingAttributeLarge() #0
; CHECK: define void @outerAttribute() {
; CHECK: attributes #0 = { "min-legal-vector-width"="512" }
; RUN: opt -consthoist -S -o - %s | FileCheck %s
target triple = "thumbv6m-none--musleabi"

; Check that for i8 type, the maximum legal offset is 31.
; Also check that an constant used as value to be stored rather than
; pointer in a store instruction is hoisted.
; CHECK: foo_i8
; CHECK-DAG:  %[C1:const[0-9]?] = bitcast i32 805874720 to i32
; CHECK-DAG:  %[C2:const[0-9]?] = bitcast i32 805874688 to i32
; CHECK-DAG:  %[C3:const[0-9]?] = bitcast i32 805873720 to i32
; CHECK-DAG:  %[C4:const[0-9]?] = bitcast i32 805873688 to i32
; CHECK:  %0 = inttoptr i32 %[C2] to i8*
; CHECK-NEXT:  %1 = load volatile i8, i8* %0
; CHECK-NEXT:  %[M1:const_mat[0-9]?] = add i32 %[[C2]], 4
; CHECK-NEXT:  %2 = inttoptr i32 %[[M1]] to i8*
; CHECK-NEXT:  %3 = load volatile i8, i8* %2
; CHECK-NEXT:  %[M2:const_mat[0-9]?] = add i32 %[[C2]], 31
; CHECK-NEXT:  %4 = inttoptr i32 %[[M2]] to i8*
define void @foo_i8() {
  entry:
  %0 = load volatile i8, i8* inttopr (i32 805874688 to i8*)
  %1 = load volatile i8, i8* inttopr (i32 805874692 to i8*)
  %2 = load volatile i8, i8* inttopr (i32 805874719 to i8*)
  %3 = load volatile i8, i8* inttopr (i32 805874720 to i8*)
  %4 = load volatile i8, i8* inttopr (i32 805874727 to i8*)
  store i8 %4, i8* inttopr(i32 805873688 to i8*)
  store i8 %3, i8* inttopr(i32 805873719 to i8*)
  store i8 %2, i8* inttopr(i32 805873720 to i8*)
  store i8 %1, i8* inttopr(i32 805873727 to i8*)
  store i8* inttopr(i32 805874800 to i8*), i8** @goo
  ret void
}

; Check that for i16 type, the maximum legal offset is 62.
; CHECK: foo_i16
; CHECK-DAG: %[[C1:const[0-9]?]] = bitcast i32 805874752 to i32
; CHECK-DAG: %[[C2:const[0-9]?]] = bitcast i32 805874688 to i32
; CHECK: %0 = inttopr i32 %[[C2]] to i16*
; CHECK-NEXT: %1 = load volatile i16, i16* %0, align 2
; CHECK-NEXT: %[[M1:const_mat[0-9]?]] = add i32 %[[C2]], 4
; CHECK-NEXT: %2 = inttopr i32 %[[M1]] to i16*
; CHECK-NEXT: %3 = load volatile i16, i16* %2, align 2
; CHECK-NEXT: %[[M2:const_mat[0-9]?]] = add i32 %[[C2]], 32
; CHECK-NEXT: %4 = inttopr i32 %[[M2]] to i16*
define void @foo_i16() {
entry:
%0 = load volatile i16, i16* inttoptr (i32 805874688 to i16*), align 2
%1 = load volatile i16, i16* inttoptr (i32 805874692 to i16*), align 2
%2 = load volatile i16, i16* inttoptr (i32 805874720 to i16*), align 2
%3 = load volatile i16, i16* inttoptr (i32 805874750 to i16*), align 2
%4 = load volatile i16, i16* inttoptr (i32 805874752 to i16*), align 2
%5 = load volatile i16, i16* inttoptr (i32 805874774 to i16*), align 2
ret void
}

define void @foo_i32() {
entry:
%0 = load volatile i32, i32* inttoptr (i32 805874688 to i32*), align 4
%1 = load volatile i32, i32* inttoptr (i32 805874692 to i32*), align 4
%2 = load volatile i32, i32* inttoptr (i32 805874720 to i32*), align 4
%3 = load volatile i32, i32* inttoptr (i32 805874750 to i32*), align 4
%4 = load volatile i32, i32* inttoptr (i32 805874752 to i32*), align 4
%5 = load volatile i32, i32* inttoptr (i32 805874774 to i32*), align 4
ret void
}
# added to the AUTHORS file, depending on whether the
# individual or corporate CLA was used.
#
# Names should be added to this file as:
#     Name <email address>
#
# Please keep the list sorted.

Albert Pretorius <pretoalb@gmail.com>
Arne Beer <arne@twobeer.de>
Billy Robert O'Neal III <billy.oneal@gmail.com> <bion@microsoft.com>
Chris Kennelly <ckennelly@google.com> <ckennelly@ckennelly.com>
Christopher Seymour <chris.j.seymour@hotmail.com>
David Coeurjolly <david.coeurjolly@liris.cnrs.fr>
Deniz Evrenci <denizevrenci@gmail.com>
Dominic Hamon <dma@stripysock.com> <dominic@google.com>
Dominik Czarnota <dominik.b.czarnota@gmail.com>
Eric Fiselier <eric@efcs.ca>
Eugene Zhuk <eugene.zhuk@gmail.com>
Evgeny Safronov <division494@gmail.com>
Felix Homann <linuxaudio@showlabor.de>
Ismael Jimenez Martinez <ismael.jimenez.martinez@gmail.com>
Jern-Kuan Leong <jernkuan@gmail.com>
JianXiong Zhou <zhoujianxiong2@gmail.com>
Joao Paulo Magalhaes <joaoppmagalhaes@gmail.com>
John Millikin <jmillikin@stripe.com>
Jussi Knuuttila <jussi.knuuttila@gmail.com>
Kai Wolf <kai.wolf@gmail.com>
Kishan Kumar <kumar.kishan@outlook.com>
Kaito Udagawa <umireon@gmail.com>
Lei Xu <eddyxu@gmail.com>
Matt Clarkson <mattyclarkson@gmail.com>
Maxim Vafin <maxvafin@gmail.com>
Nick Hutchinson <nshutchinson@gmail.com>
Oleksandr Sochka <sasha.sochnka@gmail.com>
Pascal Leroy <phl@google.com>
Paul Redmond <paul.redmond@gmail.com>
Pierre Phaneuf <pphaneuf@google.com>
Radoslav Yovchev <radoslav.tm@gmail.com>
Raul Marin <rmrodriguez@cartodb.com>
Ray Glover <ray.glover@uk.ibm.com>
Robert Guo <robert.guo@mongodb.com>
Roman Lebedev <lebedev.ri@gmail.com>
Shuo Chen <chenshuo@chenshuo.com>
Tobias Ulvgrd <tobias.ulvgard@dirac.se>
Tom Madams <tom.ej.madams@gmail.com> <tmadams@gmail.com>
Yixuan Qiu <yixuanq@gmail.com>
Yusuke Suzuki <utatane.tea@gmail.com>
Zbigniew Skowron <zbychs@gmail.com>
==============================================================================
LLVM Release License
==============================================================================
University of Illinois/NCSA
Open Source License

Copyright (c) 2003-2019 University of Illinois at Urbana-Champaign.
All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

http://llvm.org

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal with
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimers.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimers in the
documentation and/or other materials provided with the distribution.

* Neither the names of the LLVM Team, University of Illinois at
  Urbana-Champaign, nor the names of its contributors may be used to
  endorse or promote products derived from this Software without specific
  prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE
SOFTWARE.

==============================================================================
Copyrights and Licenses for Third Party Software Distributed with LLVM:
==============================================================================
The LLVM software contains code written by third parties. Such software will have its own individual LICENSE.TXT file in the directory in which it appears. This file will describe the copyrights, license, and restrictions which apply to that code.

The disclaimer of warranty in the University of Illinois Open Source License applies to all code in the LLVM Distribution, and nothing in any of the other licenses gives permission to use the names of the LLVM Team or the University of Illinois to endorse or promote products derived from this Software.

The following pieces of software have additional or alternate copyrights, licenses, and/or restrictions:

<table>
<thead>
<tr>
<th>Program</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Test</td>
<td>llvm/utils/unittest/googletest</td>
</tr>
<tr>
<td>OpenBSD regex</td>
<td>llvm/lib/Support/{reg*, COPYRIGHT.regex}</td>
</tr>
<tr>
<td>pyyaml tests</td>
<td>llvm/test/YAMLParser/{*.data, LICENSE.TXT}</td>
</tr>
<tr>
<td>ARM contributions</td>
<td>llvm/lib/Target/ARM/LICENSE.TXT</td>
</tr>
<tr>
<td>md5 contributions</td>
<td>llvm/lib/Support/MD5.cpp llvm/include/llvm/Support/MD5.h</td>
</tr>
</tbody>
</table>

The LLVM System Interface Library

The LLVM System Interface Library is licensed under the Illinois Open Source License and has the following additional copyright:

Copyright (C) 2004 eXtensible Systems, Inc.

; RUN: llc < %s -mtriple=x86_64-linux-android -mattr=+mmx -enable-legalize-types-checking | FileCheck %s ;
; D31946
; Check that we dont end up with the "LLVM ERROR: Cannot select" error.
; Additionally ensure that the output code actually put fp128 values in SSE registers.

declare fp128 @llvm.fabs.f128(fp128)
declare fp128 @llvm.copysign.f128(fp128, fp128)

define fp128 @TestSelect(fp128 %a, fp128 %b) {
    %cmp = fcmp ogt fp128 %a, %b
    %sub = fsub fp128 %a, %b
    %res = select i1 %cmp, fp128 %sub, fp128 0xL0000000000000000
    ret fp128 %res
    ; CHECK-LABEL: TestSelect:
    ; CHECK movaps 16(%rsp), %xmm1
    ; CHECK-NEXT callq _subtf3
    ; CHECK-NEXT testl %ebx, %ebx
    ; CHECK-NEXT jg .LBB0_2
    ; CHECK-NEXT # %bb.1:
    ; CHECK-NEXT movaps .LCPI0_0(rip), %xmm0
define fp128 @TestFabs(fp128 %a) {
    %res = call fp128 @llvm.fabs.f128(fp128 %a)
    ret fp128 %res
}

define fp128 @TestCopysign(fp128 %a, fp128 %b) {
    %res = call fp128 @llvm.copysign.f128(fp128 %a, fp128 %b)
    ret fp128 %res
}

define fp128 @TestFneg(fp128 %a) {
    %mul = fmul fp128 %a, %a
    %res = fsub fp128 0xL00000000000000008000000000000000, %mul
    ret fp128 %res
}

ARM Limited

Software Grant License Agreement ("Agreement")

Except for the license granted herein to you, ARM Limited ("ARM") reserves all right, title, and interest in and to the Software (defined below).

Definition

"Software" means the code and documentation as well as any original work of authorship, including any modifications or additions to an existing work, that is intentionally submitted by ARM to llvm.org (http://llvm.org) ("LLVM") for inclusion in, or documentation of, any of the products owned or managed by LLVM (the "Work"). For the purposes of this definition, "submitted" means any form of
electronic, verbal, or written communication sent to LLVM or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, LLVM for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked otherwise.

1. Grant of Copyright License. Subject to the terms and conditions of this Agreement, ARM hereby grants to you and to recipients of the Software distributed by LLVM a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, sublicense, and distribute the Software and such derivative works.

2. Grant of Patent License. Subject to the terms and conditions of this Agreement, ARM hereby grants you and to recipients of the Software distributed by LLVM a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by ARM that are necessarily infringed by ARM's Software alone or by combination of the Software with the Work to which such Software was submitted. If any entity institutes patent litigation against ARM or any other entity (including a cross-claim or counterclaim in a lawsuit) alleging that ARM's Software, or the Work to which ARM has contributed constitutes direct or contributory patent infringement, then any patent licenses granted to that entity under this Agreement for the Software or Work shall terminate as of the date such litigation is filed.

Unless required by applicable law or agreed to in writing, the software is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.658 commons-lang3 3.8.1
1.658.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.659 math-random 1.0.4

1.659.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.660 animal-sniffer-annotation 1.18
1.661 homedir-polyfill 1.0.3 1.0.3

1.661.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Brian Woodward

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.


1.662.1 Available under license:

MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.663 depd 1.1.0 1.1.0

1.663.1 Available under license :
(The MIT License)

Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.664 string_decoder 1.0.3 1.0.3

1.664.1 Available under license :

Node.js is licensed for use as follows:

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.665 isobject 3.0.1 3.0.1

1.665.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.666 object-keys 1.0.12
1.666.1 Available under license:
The MIT License (MIT)

Copyright (C) 2013 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.667 resolve-from 1.0.1
1.667.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.668 string-prototype-trimstart 1.0.4

1.668.1 Available under license:
MIT License

Copyright (c) 2017 Khaled Al-Ansari

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.669 braces 2.3.2 2.3.2
1.669.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.670 object-defaults 1.1.0 1.1.0

1.670.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.671 commons-math 2.2
1.671.1 Available under license:

Apache Commons Math
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

===============================================================================
The BracketFinder (package org.apache.commons.math.optimization.univariate) and PowellOptimizer (package org.apache.commons.math.optimization.general) classes are based on the Python code in module "optimize.py" (version 0.5) developed by Travis E. Oliphant for the SciPy library (http://www.scipy.org/) Copyright © 2003-2009 SciPy Developers.
===============================================================================
The LinearConstraint, LinearObjectiveFunction, LinearOptimizer, RelationShip, SimplexSolver and SimplexTableau classes in package org.apache.commons.math.optimization.linear include software developed by Benjamin McCann (http://www.benmccann.com) and distributed with the following copyright: Copyright 2009 Google Inc.
===============================================================================
This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.
The LevenbergMarquardtOptimizer class in package org.apache.commons.math.optimization.general includes software translated from the lmder, lmpar and qrsolv Fortran routines from the Minpack package Minpack Copyright Notice (1999) University of Chicago. All rights reserved
===============================================================================
The GraggBulirschStoerIntegrator class in package org.apache.commons.math.ode.nonstiff includes software translated from the odex Fortran routine developed by E. Hairer and G. Wanner. Original source copyright:
Copyright (c) 2004, Ernst Hairer
===============================================================================
The EigenDecompositionImpl class in package org.apache.commons.math.linear includes software translated from some LAPACK Fortran routines. Original source copyright:
Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.
The MersenneTwister class in package org.apache.commons.math.random includes software translated from the 2002-01-26 version of the Mersenne-Twister generator written in C by Makoto Matsumoto and Takuji Nishimura. Original source copyright:
Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,
All rights reserved

The complete text of licenses and disclaimers associated with the the original sources enumerated above at the time of code translation are in the LICENSE.txt file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERM AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS MATH DERIVATIVE WORKS:

The Apache commons-math library includes a number of subcomponents whose implementation is derived from original sources written in C or Fortran. License terms of the original sources are reproduced below.

===============================================================================
For the lmderr, lmpar and qrsolv Fortran routine from minpack and translated in the LevenbergMarquardtOptimizer class in package org.apache.commons.math.optimization.general
Original source copyright and license statement:

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

_______________________________________________________________
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

===============================================================================

Copyright and license statement for the odex Fortran routine developed by E. Hairer and G. Wanner and translated in GraggBulirschStoerIntegrator class in package org.apache.commons.math.ode.nonstiff:
Copyright (c) 2004, Ernst Hairer

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================
Copyright and license statement for the original lapack fortran routines
translated in EigenDecompositionImpl class in package
org.apache.commons.math.linear:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

$COPYRIGHT$

Additional copyrights may follow

$HEADERS$

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials
- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

Copyright and license statement for the original Mersenne twister C routines translated in MersenneTwister class in package org.apache.commons.math.random:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
1.672 nspr 4.13.1 0ubuntu0.16.04.1
1.672.1 Available under license :
Mozilla Public License Version 2.0
==================================
1. Definitions
--------------

1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
   in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
   version 1.1 or earlier of the License, but not also under the
   terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
   means this document.

1.9. "Licensable"
   means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
   means any of the following:

   (a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

   (b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
   means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
   means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
   means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
   means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions
2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses
No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter
the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.
5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis,*
*  without warranty of any kind, either expressed, implied, or statutory, *
*  including, without limitation, warranties that the Covered Software *
*  is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the *
*  quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You *
*  (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer. *
*                                                                      *
************************************************************************
7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions
Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
---------------------------------------------
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------
This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
1.673 jsreport-authorization 2.4.0 2.4.0

1.673.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a
facility refers to a function or data to be supplied by an Application
that uses the facility (other than as an argument passed when the
facility is invoked), then you may convey a copy of the modified
version:

a) under this License, provided that you make a good faith effort to
ensure that, in the event an Application does not supply the
function or data, the facility still operates, and performs
whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of
this License applicable to that copy.


The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:
a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.674 esutils 2.0.3 2.0.3
1.674.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
1.675 yauaa-logparser 5.20

1.675.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Yet Another UserAgent Analyzer
 * Copyright (C) 2013-2020 Niels Basjes
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/path/to/permission/1202446063_1631568159.8/yauaa-logparser-5-20-sources.jar/nl/basjes/parse/useragent/dissector/UserAgentDissector.java

No license file was found, but licenses were detected in source scan.

2020 Niels Basjes

~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1202446063_1631568159.8/0/yauaa-logparser-5-20-sources-jar/META-INF/maven/nl.basjes.parse.useragent/yauaa-logparser/pom.xml

1.676 @babel/helper-member-expression-to-functions 7.16.0
1.676.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.677 prelude-ls 1.1.2 1.1.2
1.677.1 Available under license :
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.678 protobuf-java 3.11.1

1.678.1 Available under license:

<OWNER> = Regents of the University of California
<ORGANIZATION> = University of California, Berkeley
<YEAR> = 1998

In the original BSD license, both occurrences of the phrase "COPYRIGHT HOLDERS AND CONTRIBUTORS" in the disclaimer read "REGENTS AND CONTRIBUTORS".

Here is the license template:

Copyright (c) <YEAR>, <OWNER>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.679 strip-eof 1.0.0 1.0.0

1.679.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.680 anymatch 1.3.2

1.680.1 Available under license:
The ISC License

Copyright (c) 2014 Elan Shanker
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.681 @babel/runtime 7.12.1 7.12.1

1.681.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.682 lodash.once 4.1.1 4.1.1

1.682.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.683 shebang-command 2.0.0

1.683.1 Available under license:

MIT License

Copyright (c) Kevin Mrtensson <kevinmartessson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.684 glob-parent 5.1.2

1.684.1 Available under license:

The ISC License

Copyright (c) 2015, 2019 Elan Shanker
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.685 extglob 2.0.4 2.0.4
1.685.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.686 jsp 2.3.1
1.686.1 Available under license :
   12313989
   DEPLOYED
   1.1
   Common Development and Distribution License (CDDL)
   Details
   License Type: OPENSOURCE
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof),
with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third
party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version
of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding
declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.
8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ’252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise
dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version,
or (ii) the combination of Modifications made by that
Contributor with other software (except as part of the
Contributor Version) or other devices; or (3) under Patent
Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make
available in Executable form must also be made available in Source
Code form and that Source Code form must be distributed only under
the terms of this License. You must include a copy of this License
with every copy of the Source Code form of the Covered Software
You distribute or otherwise make available. You must inform
recipients of any such Covered Software in Executable form as to
how they can obtain such Covered Software in Source Code form in a
reasonable manner on or through a medium customarily used for
software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are
governed by the terms of this License. You represent that You
believe Your Modifications are Your original creation(s) and/or
You have sufficient rights to grant the rights conveyed by this
License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that
identifies You as the Contributor of the Modification. You may not
remove or alter any copyright, patent or trademark notices
contained within the Covered Software, or any notices of licensing
or any descriptive text giving attribution to any Contributor or
the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in
Source Code form that alters or restricts the applicable version
of this License or the recipients' rights hereunder. You may
choose to offer, and to charge a fee for, warranty, support,
indemnity or liability obligations to one or more recipients of
Covered Software. However, you may do so only on Your own behalf,
and not on behalf of the Initial Developer or any Contributor. You
must make it absolutely clear that any such warranty, support,
indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

[-Sun Microsystems, Inc.-]

{+Oracle+} is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software.
If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are anInitial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where
the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. {+If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4.+} In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR [-LOST PROFITS,-] LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is
defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
1.687 dns-prefetch-control 0.1.0 0.1.0

1.687.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Evan Hahn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.688 pseudomap 1.0.2 1.0.2

1.688.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.690 jackson-mapper-asl 1.9.13
1.690.1 Available under license:
   No license file was found, but licenses were detected in source scan.

    /* Jackson JSON-processor.
    *   Copyright (c) 2007- Tatu Saloranta, tatu.saloranta@iki.fi
    *   Licensed under the License specified in file LICENSE, included with
    *   the source code and binary code bundles.
    *   You may not use this file except in compliance with the License.
    *   Unless required by applicable law or agreed to in writing, software
    *   distributed under the License is distributed on an "AS IS" BASIS,
    *   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
    *   See the License for the specific language governing permissions and
    *   limitations under the License.
    */

    Found in path(s):
    */opt/ws_local/PERMITS_SQL/1078286973_1597285663.32/0/jackson-mapper-asl-1-9-13-sources-2-
    jar/org/codehaus/jackson/map/MappingJsonFactory.java

1.691 babel-helper-regex 6.26.0 6.26.0
1.691.1 Available under license:
The MIT License (MIT)

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
1.692 ansi-styles 2.2.1 2.2.1

1.692.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.693 hive-service-rpc 3.1.2

1.693.1 Available under license:

Hive Service RPC
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and eachContributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.694 make-iterator 1.0.1 1.0.1

1.694.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.695 reactive-streams 1.0.2

1.695.1 Available under license:

No license file was found, but licenses were detected in source scan.

/************************************************************************
 */ Licensed under Public Domain (CC0)                              */
 */ * To the extent possible under law, the person who associated CC0 with */
 */ * this code has waived all copyright and related or neighboring   */
 */ * rights to this code.                                             */
 */ * You should have received a copy of the CC0 legal code along with this */
 */ * work. If not, see <http://creativecommons.org/publicdomain/zero/1.0/>.*/
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092915250_1600810885.15/0/reactive-streams-1-0-2-sources-1-jar/org/reactivestreams/Subscriber.java
* /opt/ws_local/PERMITS_SQL/1092915250_1600810885.15/0/reactive-streams-1-0-2-sources-1-jar/org/reactivestreams/Subscription.java
* /opt/ws_local/PERMITS_SQL/1092915250_1600810885.15/0/reactive-streams-1-0-2-sources-1-jar/org/reactivestreams/Publisher.java
* /opt/ws_local/PERMITS_SQL/1092915250_1600810885.15/0/reactive-streams-1-0-2-sources-1-jar/org/reactivestreams/Processor.java

1.696 bluebird 3.5.4 3.5.4
1.696.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013-2018 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.697 fast-csv 3.7.0 3.7.0
1.697.1 Available under license:
The MIT License

Copyright (c) 2011-2019 C2FO

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.698 sketches-core 0.9.0

1.698.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-16, Yahoo! Inc.
 * Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root for terms.
 */
/**
 * The MurmurHash3 is a fast, non-cryptographic, 128-bit hash function that has
 * excellent avalanche and 2-way bit independence properties.
 */
*
*
* The C++ MurmurHash3_x64_128(...), revision 150, of the MurmurHash3, written by Austin Appleby,
* which is in the Public Domain, was the inspiration for this implementation in Java.
* The C++ version can be found at
* <a href= "https://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp"> SMHasher &amp;
* MurmurHash</a>.
* </p>
*
*
* This java implementation pays close attention to the C++ algorithms in order to
* maintain bit-wise compatibility, but the design is quite different. This implementation has also
* been extended to include processing of arrays of longs, char or ints, which was not part of the
* original C++ implementation. This implementation produces the same exact output hash bits as
* the above C++ method given the same input.</p>
*
* In addition, with this implementation, the hash of byte[], char[], int[], or long[] will
* produce the same hash result if, and only if, all the arrays have the same exact length in
* bytes, and if the contents of the values in the arrays have the same byte endianness and
* overall order. There is a unit test for this class that demonstrates this.</p>
*
* The structure of this implementation also reflects a separation of code that is dependent on the
* input structure (in this case byte[], int[] or long[]) from code that is independent of the input
* structure. This also makes the code more readable and suitable for future extensions.
* </p>
*
* @author Lee Rhodes
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/hash/MurmurHash3.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2016, Yahoo! Inc. Licensed under the terms of the
* Apache License 2.0. See LICENSE file at the project root for terms.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/quantiles/DoublesMergeImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/theta/IntersectionImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/quantiles/DoublesUpdateImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/quantiles/DoublesPmfCdfImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/theta/PairwiseSetOperations.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/quantiles/DirectCompactDoublesSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/quantiles/DirectUpdateDoublesSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/hll/CouponsIterator.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/hll/HllMap.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/hll/CouponHashMap.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
jar/com/yahoo/sketches/hll/SingleCouponMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016, Yahoo! Inc.
 * Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root
 * for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-17, Yahoo! Inc.
 * Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/PreambleUtil.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/ReservoirLongsUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/SamplingUtil.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/ReservoirItemsUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/ReservoirSize.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/VersionConverter.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/ReservoirItemsSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/ReservoirLongsSketch.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016, Yahoo! Inc.
* Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root
* for terms.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-17, Yahoo! Inc.
 * Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root
 * for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/sampling/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016, Yahoo! Inc.
 * Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/ErrorType.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/Util.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsUtil.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/PreambleUtil.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017, Yahoo! Inc. Licensed under the terms of the
 * Apache License 2.0. See LICENSE file at the project root for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-
No license file was found, but licenses were detected in source scan.

Copyright 2015, Yahoo! Inc.

Licensed under the terms of the Apache License 2.0.

See LICENSE file at the project root for terms.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsPmfCdfImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsMergeImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsByteArrayImpl.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsUpdateImpl.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2015-16, Yahoo! Inc.
* Licensed under the terms of the Apache License 2.0. See LICENSE file at the project root for terms.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/package-info.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesAuxiliary.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/ArrayOfDoublesUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/ArrayOfDoublesUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/ArrayOfDoublesAnotB.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/Intersection.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/QuickSelect.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/QuickSelect.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/QuickSelect.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/QuickSelect.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hll/Preamble.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hll/DenseCompressedFieldsFactory.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/ArrayOfDoublesAnotB.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hll/CompositeBucketIterator.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/SketchesArgumentException.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/Sketches.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/HeapArrayOfDoublesCompactSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesUtil.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hll/Interpolation.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hll/DenseFieldsFactory.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/Union.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/tuple/UpdatableSummary.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/UpdateReturnState.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/HeapArrayOfDoublesQuickSelectSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/FieldsFactory.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/SketchesStateException.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/QuickSelectSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/ArrayOfType16StringsSerDe.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/ArrayOfNumbersSerDe.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/SketchesStateException.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/theta/SetOperation.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hash/MurmurHash3Adaptor.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hash/package-info.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hash/UpdatableHashSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/hash/MurmurHash3Adaptor.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/ItemsAuxiliary.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/UpdatableHashSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesUnion.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/DoublesUtil.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/UpdatableHashSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/SketchesStateException.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/SketchesStateException.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/quantiles/Ordering.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016, Yahoo! Inc. Licensed under the terms of the Apache License 2.0. See LICENSE file
 * at the project root for terms.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/LongsSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/ReversePurgeItemHashMap.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/ReversePurgeLongHashMap.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/ItemsSketch.java
* /opt/ws_local/PERMITS_SQL/1066571456_1593690082.73/0/sketches-core-0-9-0-sources-jar/com/yahoo/sketches/frequencies/PreambleUtil.java

Open Source Used In DNAC Platform Arica 2532
1.699 babel-plugin-transform-es2015-modules-systemjs 6.24.1

1.699.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrits noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION...
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.700 httpsnippet 1.16.5 1.16.5

1.700.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Mashape (https://www.mashape.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.701 ansi-styles 3.2.1

1.701.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.702 opentype-js 0.8.0 0.8.0

1.702.1 Available under license:
The MIT License (MIT)

Copyright (c) 2017 Frederik De Bleser

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.703 jersey-server 2.32

1.703.1 Available under license:

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

**Angular JS, v1.6.6**
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

**aopalliance Version 1**
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

**Bean Validation API 2.0.2**
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

**Hibernate Validator CDI, 6.1.2.Final**
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

**Bootstrap v3.3.7**
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/
# Notice for Jersey Core Server module
This content is produced and maintained by the Eclipse Jersey project.

* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
   where such changes and/or additions to the Program originate from
   and are Distributed by that particular Contributor. A Contribution
   "originates" from a Contributor if it was added to the Program by
   such Contributor itself or anyone acting on such Contributor's behalf.
   Contributions do not include changes or additions to the Program that
   are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity.
Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements
of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance
claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost
of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey

---

Open Source Used In DNAC Platform Arica 2552
the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

### 1.704 unescape 1.0.1 1.0.1

#### 1.704.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014, 2016-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.705 @babel/plugin-syntax-class-static-block 7.14.5

1.705.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.706 java-jwt 3.8.3

1.706.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.707 pure-color 1.3.0 1.3.0

1.707.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Nick Williams
Copyright (c) 2011 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.708 js-tokens 3.0.2 3.0.2

1.708.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.709 jersey-bean-validation 2.5.1

1.710 zip-stream 1.2.0 1.2.0

1.710.1 Available under license:
Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.711 cacheable-request 6.1.0
1.711.1 Available under license:
MIT License

Copyright (c) 2017 Luke Childs

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.712 geo-ip 2.14.0
1.712.1 Available under license:
No license file was found, but licenses were detected in source scan.

    public DomainResponse(
    public DomainResponse(

Found in path(s):
* /opt/cola/permits/1140689171_1614870592.56/0/geoip2-2-14-0-sources-1-jar/com/maxmind/geoip2/model/DomainResponse.java

No license file was found, but licenses were detected in source scan.

    <name>Apache License, Version 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>

Found in path(s):
* /opt/cola/permits/1140689171_1614870592.56/0/geoip2-2-14-0-sources-1-jar/META-INF/maven/com.maxmind/geoip2/geoip2/pom.xml

No license file was found, but licenses were detected in source scan.
public DomainResponse domain(InetAddress ipAddress) throws IOException,

Found in path(s):
* /opt/cola/permits/1140689171_1614870592.56/0/geoip2-2-14-0-sources-1-
jar/com/maxmind/geoip2/DatabaseReader.java

1.713 kerb-common 1.0.0
1.713.1 Available under license :

Kerby-kerb Common
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.714 html-to-xlsx 2.0.3 2.0.3

1.714.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Jan Blaha

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.715 yargs 16.2.0 16.2.0

1.715.1 Available under license :

MIT License

Copyright 2010 James Halliday (mail@substack.net); Modified work Copyright 2014 Contributors (ben@npmjs.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.716 minimist 0.0.8 0.0.8

1.716.1 Available under license :

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.717 lodash-groupby 4.6.0 4.6.0

1.717.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

**1.718 big-integer 1.6.48 1.6.48**

*1.718.1 Available under license :*

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org>

**1.719 regexpu-core 2.0.0 2.0.0**

*1.719.1 Available under license :*

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.720 mv 2.1.1 2.1.1

1.720.1 Available under license:
Copyright (c) 2014 Andrew Kelley

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.721 winston 2.4.5 2.4.5
1.721.1 Available under license:

Copyright (c) 2010 Charlie Robbins

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.722 netty-transport 4.1.59.Final

1.722.1 Available under license:

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/cola/permits/1145449543_1616156406.95/0/netty-transport-4-1-59-final-1-jar/META-INF/native-image/io.netty/transport/native-image.properties
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Implementation-Title: Netty/Transport
Bundle-Description: Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers and clients.
Automatic-Module-Name: io.netty.transport
Bundle-License: https://www.apache.org/licenses/LICENSE-2.0
Bundle-SymbolicName: io.netty.transport
Implementation-Version: 4.1.59.Final
Built-By: root
Bnd-LastModified: 1612782105444
Open Source Used In DNAC Platform Arica

Bundle-ManifestVersion: 2
Implementation-Vendor-Id: io.netty
Bundle-DocURL: https://netty.io/
Bundle-Vendor: The Netty Project
Import-Package: io.netty.buffer;version="[4.1,5)";io.netty.resolver;version="[4.1,5)";io.netty.util;version="[4.1,5)";io.netty.util.concurrent;version="[4.1,5)";io.netty.util.internal;version="[4.1,5)";io.netty.util.internal.logging;version="[4.1,5)";sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="[1,2)";resolution:=optional
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.6))"
Tool: Bnd-2.4.1.201501161923
Implementation-Vendor: The Netty Project
Bundle-Name: Netty/Transport
Bundle-Version: 4.1.59.Final
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_252
Implementation-URL: https://netty.io/netty-transport/

Found in path(s):
* /opt/cola/permits/1145449543_1616156406.95/0/netty-transport-4-1-59-final-1-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0

---
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1145449543_1616156406.95/0/netty-transport-4-1-59-final-1-jar/META-INF/maven/io.netty/netty-transport/pom.xml

1.723 jsreport-studio-theme-dark 0.3.0 0.3.0
1.723.1 Available under license :
MIT License

Copyright (c) 2019 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.724 jsesc 2.5.2
1.724.1 Available under license :
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.725 drill-format-pcapng 1.19.0
1.725.1 Available under license:

Drill : Contrib : Format : Pcap-NG
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided...
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.726 node-promise 0.5.14 0.5.14

1.726.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.727 byte-buddy-agent 1.9.3
1.727.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 - 2018 Rafael Winterhalter
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1114503625_1607518978.69/0/byte-buddy-agent-1-9-3-sources-1-jar/net/bytebuddy/agent/Installer.java
* /opt/cola/permits/1114503625_1607518978.69/0/byte-buddy-agent-1-9-3-sources-1-jar/net/bytebuddy/agent/package-info.java
* /opt/cola/permits/1114503625_1607518978.69/0/byte-buddy-agent-1-9-3-sources-1-jar/net/bytebuddy/agent/Attacher.java
* /opt/cola/permits/1114503625_1607518978.69/0/byte-buddy-agent-1-9-3-sources-1-jar/net/bytebuddy/agent/ByteBuddyAgent.java
* /opt/cola/permits/1114503625_1607518978.69/0/byte-buddy-agent-1-9-3-sources-1-jar/net/bytebuddy/agent/VirtualMachine.java
**1.728 uritemplate 0.3.4 0.3.4**

**1.728.1 Available under license:**

No license file was found, but licenses were detected in source scan.

```javascript
/*global unescape, module, define, window, global*/

/*
* UriTemplate Copyright (c) 2012-2013 Franz Antesberger. All Rights Reserved.
* Available via the MIT license.
*/

(function (exportCallback) {
    "use strict";

    Found in path(s):
    * /opt/ws_local/PERMITS_SQL/1063105573_1593668271.03/0/uritemplate-js-0-3-4-zip/uritemplate-js-0.3.4/src/pre.txt
    No license file was found, but licenses were detected in source scan.

    /*
    * UriTemplate Copyright (c) 2012-2013 Franz Antesberger. All Rights Reserved.
    * Available via the MIT license.
    */

    Found in path(s):
    * /opt/ws_local/PERMITS_SQL/1063105573_1593668271.03/0/uritemplate-js-0-3-4-zip/uritemplate-js-0.3.4/bin/uritemplate.js
    No license file was found, but licenses were detected in source scan.

    URI Template JS
    ===============

    This is a javascript implementation of [RFC6570](http://tools.ietf.org/html/rfc6570) - URI Template, and can expand templates up to and including Level 4 in that specification.

    It exposes a constructor function UriTemplate with the two methods:

    * (static) parse(uriTemplateText) returning an instance of UriTemplate
    * expand(variables) returning an string

    Be aware, that a parsed UriTemplate is frozen, so it is stateless. You can reuse instances of UriTemplates.

    Requirements
    ============

    You can use uritemplate.js in any even not so modern browser (Tested even with IE8 in IE7-Mode), see file
**npm install uritemplate**

And then in a node application:

```javascript
var
    UriTemplate = require('uritemplate'),
    template;

template = UriTemplate.parse('{?query*}');
template.expand({query: {first: 1, second: 2}});
--> "?first=1&second=2"
```

Or within a html document (see also demo.html):

```html
<script type="text/javascript" src="bin/uritemplate.js"></script>
<script type="text/javascript">
    var template = UriTemplate.parse('{?query*}');
    alert(template.expand({query: {first: 1, second: 2}}));
</script>
```

If you want to clone the git project, be aware of the submodule uritemplate-test. So you have to:

```bash
git clone --recursive https://github.com/fxa/uritemplate-js.git
```

**Build**

```bash
npm install
npm test
```

**Tests**

The integration tests are taken from https://github.com/uri-templates/uritemplate-test as a submodule. The tests are integrated in the Jakefile.

**License**

Copyright 2013 Franz Antesberger

MIT License, see http://mit-license.org/

**Release Notes**
* 0.3.3 fixed https://github.com/ fx a/uritemplate-js/ issues/12 Pct-Encoding-Error with chars < 0x100 (missing 0-paddin g). Thanks to https:// github.com/pfa ze!
* 0.3.2 fixed https://github.com/ fxa/uritemplate-js/ issues/11 Problems with older IE versions. Thanks to anozaki!
* 0.3.1 fixed https://github.com/ fxa/uritemplate-js/ issues/10 thank you, Paul-Martin!
* 0.3.0 introduced UriTemplateError as exception, so the interface changed from string to UriTemplateError (as the rfc suggested)
* 0.2.4 fixed double encoding according [RubenVerborgh] and some Prefix modifiers bugs
* 0.2.3 fixed bug with empty objects ('{?empty}' with '{empty:[{}]}’ shall expand to ?empty=')
* 0.2.2 fixed pct encoding bug with multibyte utf8 chars
* 0.2.1 fixed a bug in package.json
* 0.2.0 heavy project refactoring, splitting source files, introducing jshint (preparation of next steps)

Next Steps
---------
* Implementing more unit tests (now only a view tests are implemented)
* A new method extract(uri), which tries to extract the variables from a given uri.
This is harder, than you might think

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1063105573_1593668271.03/0/uritemplate-js-0-3-4-zip/uritemplate-js-0.3.4/README.md
No license file was found, but licenses were detected in source scan.

 require.js 0.27.1 Copyright (c) 2010-2011, The Dojo Foundation All Rights Reserved.
Available via the MIT or new BSD license.
see: http://github.com/jrburke/requirejs for details

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1063105573_1593668271.03/0/uritemplate-js-0-3-4-zip/uritemplate-js-0.3.4/require.js

1.729 @babel/plugin-transform-async-to-generator 7.16.0

1.729.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.730 socket.io-client 2.3.0 2.3.0
1.730.1 Available under license :
   The MIT License (MIT)

     Copyright (c) 2014 Guillermo Rauch

     Permission is hereby granted, free of charge, to any person obtaining a copy
     of this software and associated documentation files (the "Software"), to deal
     in the Software without restriction, including without limitation the rights
     to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
     copies of the Software, and to permit persons to whom the Software is
     furnished to do so, subject to the following conditions:

     The above copyright notice and this permission notice shall be included in
     all copies or substantial portions of the Software.

     THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
     IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
     FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
     AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
     LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
     OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
     THE SOFTWARE.

1.731 indent-string 4.0.0
1.731.1 Available under license :
   MIT License

     Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.732 packaging 16.8
1.732.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
This software is made available under the terms of *either* of the licenses found in LICENSE.APACHE or LICENSE.BSD. Contributions to this software is made under the terms of *both* these licenses.

Copyright (c) Donald Stufft and individual contributors.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.733 netty 4.1.53.Final
1.733.1 Available under license :
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding
those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.734 depd 1.1.1 1.1.1

1.734.1 Available under license :
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.735 jsreport-tags 2.5.0 2.5.0

1.735.1 Available under license:

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.736 normalize-path 2.1.1 2.1.1

1.736.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.737 pirates 4.0.1

1.737.1 Available under license:
MIT License

Copyright (c) 2016-2018 Ari Porad

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.738 os-homedir 1.0.2 1.0.2

1.738.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.739 side-channel 1.0.4 1.0.4

1.739.1 Available under license:
MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.740 xtend 4.0.1 4.0.1
1.740.1 Available under license :

Copyright (c) 2012-2014 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.741 immer 8.0.1 8.0.1
1.741.1 Available under license :

MIT License

Copyright (c) 2017 Michel Weststrate

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.742 should-type 1.4.0 1.4.0

1.742.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.743 is-typedarray 1.0.0 1.0.0

1.743.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.744 has-tostringtag 1.0.0
1.744.1 Available under license :
MIT

1.745 mem 4.3.0 4.3.0
1.745.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.746 regenerator-runtime 0.13.3 0.13.3
1.746.1 Available under license :
MIT License

Copyright (c) 2014-present, Facebook, Inc.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.747 hadoop-yarn-api 3.2.2

1.747.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:

* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4.

which has the following notices:

The Netty Project

Please visit the Netty web site for more information:
Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
  * HOMEPAGE:
    * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
    * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
  * HOMEPAGE:
    * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
  * HOMEPAGE:
    * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging
facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh
Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source
Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2
AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue.
It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-
    actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression
and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct
the suffix array and the Burrows-Wheeler transformed string for any input string of
a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/
This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/
This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/
This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/
The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:
* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
  Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
* TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of Jetty :: Http Utility 9.3.19., Jetty :: IO Utility 9.3.19., Jetty :: Security 9.3.19., Jetty :: Server Core 9.3.19., Jetty :: Servlet Handling 9.3.19., Jetty :: Utilities 9.3.19., Jetty :: Utilities :: Ajax, Jetty :: Webapp Application Support 9.3.19., Jetty :: XML utilities 9.3.19., which has the following notices:
* ==============
Jetty Web Container
Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
==================================================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following
artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC Runtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* ===================================================================================================================
  == NOTICE file corresponding to section 4(d) of the Apache License, ==
  == Version 2.0, in this case for the Apache Xerces Java distribution. ==
Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0
--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed. The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0. The license is bundled with the download available here: http://www.vectorportal.com/subcategory/205/KILLER-WHALE-FREE-VECTOR.eps/iFile/9136/detailtest.asp

This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--

This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0, which has the following notices:

Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright: Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.
The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:

# Copyright 2008-2015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 2007-2008 Robert Niederreiter, Jens Klein
# Copyright 2004-2005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:

* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

  The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:

* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:

* ==============================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  ==============================================================

  This product includes software developed by
  Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:

* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:

* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:

* Copyright 2016 SmartBear Software
The binary distribution of this product bundles binaries of metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * - Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008, 2009, 2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/*++

* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
*
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html
*
* Abstract: The main routine
*
-*-*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/[lz4.h,lz4.c,lz4hc.h,lz4hc.c],

/*

LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c
*/

For hadoop-common-project/hadoop-common/src/main/native/gtest

--------------------------------------------------------------------------------

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1
--------------------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

--------------------------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@cit.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=============  

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

--------------------------------------------------------------------------------

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

--------------------------------------------------------------------------------

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the
following license:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
em-helpers v0.5.13
em-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.13.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0

Open Source Used In DNAC Platform Arica 2620
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodeings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
------------------------------------------------------------------------
Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this
file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the
License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
express or implied. See the License for the specific language governing permissions and
limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT
LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js
------------------------------------------------------------------------
The MIT License
Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery
- Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
- Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

For:
oj! Algorithms - version 43.0

--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

All files located in the node_modules and external directories are
externally maintained libraries used by this software which have their
own licenses; we recommend you read them, as their terms may differ from
the terms above.

For:
Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQLDB Database 2.3.4

(HSQL License)
"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPersonic SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group.”

The binary distribution of this product bundles these dependencies under the following license:

Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. **License** means this document.

1.8. **Licensable** means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. **Modifications** means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. **Original Software** means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. **Patent Claims** means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. **Source Code** means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. **You (or Your)** means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. **License Grants.**

2.1. **The Initial Developer Grant.** Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of
Original Software, to make, have made, use, practice, sell, and offer for sale,
and/or otherwise dispose of the Original Software (or portions thereof);

c) The licenses granted in Sections 2.1(a) and (b) are
effective on the date Initial Developer first distributes or otherwise makes the
Original Software available to a third party under the terms of this License;

d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1)for code that You delete from the Original Software, or (2)for
infringements caused by: (i) the modification of the Original Software, or
(ii) the combination of the Original Software with other software or
devices.

2.2. Contributor Grant. Conditioned upon Your compliance with
Section 3.1 below and subject to third party intellectual property claims, each
Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:

(a) under intellectual property rights (other than patent or
trademark) Licensable by Contributor to use, reproduce, modify, display,
perform, sublicense and distribute the Modifications created by such Contributor
(or portions thereof), either on an unmodified basis, with other Modifications,
as Covered Software and/or as part of a Larger Work; and

(b) under Patent
Claims infringed by the making, using, or selling of Modifications made by that
Contributor either alone and/or in combination with its Contributor Version (or
portions of such combination), to make, use, sell, offer for sale, have made,
and/or otherwise dispose of: (1) Modifications made by that Contributor (or
portions thereof); and (2) the combination of Modifications made by that
Contributor with its Contributor Version (or portions of such
combination).

c) The licenses granted in Sections 2.2(a) and 2.2(b) are
effective on the date Contributor first distributes or otherwise makes the
Modifications available to a third party.

d) Notwithstanding Section 2.2(b)
above, no patent license is granted: (1) for any code that Contributor has
deleted from the Contributor Version; (2) for infringements caused by:
(i) third party modifications of Contributor Version, or (ii) the combination
of Modifications made by that Contributor with other software (except as part of
the Contributor Version) or other devices; or (3) under Patent Claims infringed
by Covered Software in the absence of Modifications made by that
3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ are...
from this License are offered by You alone, not by the Initial Developer or
Contributor. You hereby agree to indemnify the Initial Developer and every
Contributor for any liability incurred by the Initial Developer or such
Contributor as a result of any such terms You offer.

3.6. Larger Works. You
may create a Larger Work by combining Covered Software with other code not
governed by the terms of this License and distribute the Larger Work as a single
product. In such a case, You must make sure the requirements of this License are
fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may
publish revised and/or new versions of this License from time to time. Each
version will be given a distinguishing version number. Except as provided in
Section 4.3, no one other than the license steward has the right to modify this
License.

4.2. Effect of New Versions. You may always continue to use,
distribute or otherwise make the Covered Software available under the terms of
the version of the License under which You originally received the Covered
Software. If the Initial Developer includes a notice in the Original Software
prohibiting it from being distributed or otherwise made available under any
subsequent version of the License, You must distribute and make the Covered
Software available under the terms of the version of the License under which You
originally received the Covered Software. Otherwise, You may also choose to use,
distribute or otherwise make the Covered Software available under the terms of
any subsequent version of the License published by the license
steward.

4.3. Modified Versions. When You are an Initial Developer and You
want to create a new license for Your Original Software, You may create and use
a modified version of this License if You: (a)rename the license and remove
any references to the name of the license steward (except to note that the
license differs from this License); and (b)otherwise make it clear that the
license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS,
WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT
LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS,
MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY
COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER
OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR
CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.
8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such
combination), to make, use, sell, offer for sale, have made, and/or otherwise
dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or
otherwise make available in Executable form must also be made available in
Source Code form and that Source Code form must be distributed only under the
terms of this License. You must include a copy of this License with every copy
of the Source Code form of the Covered Software You distribute or otherwise make
available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a
reasonable manner on or through a medium customarily used for software
exchange.
3.2. Modifications.
The Modifications that You create or to which
You contribute are governed by the terms of this License. You represent that You
believe Your Modifications are Your original creation(s) and/or You have
sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that
identifies You as the Contributor of the Modification. You may not remove or
alter any copyright, patent or trademark notices contained within the Covered
Software, or any notices of licensing or any descriptive text giving attribution
to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source
Code form that alters or restricts the applicable version of this License or the
recipients' rights hereunder. You may choose to offer, and to charge a fee for,
warrranty, support, indemnity or liability obligations to one or more recipients
of Covered Software. However, you may do so only on Your own behalf, and not on
behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any
7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that
the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
special, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04
--------------------------------------------------------------------------------

By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation,
with or without modification, for any purpose and without fee or royalty is
hereby granted, provided that you include the following on ALL copies of the
software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the
redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and
conditions. If none exist, the W3C Software Short Notice should be included
(hypertext is preferred, text is permitted) within the body of any redistributed
or derivative code.
- Notice of any changes or modifications to the files, including the date changes
were made. (We recommend you provide URIs to the location from which the code is
derived.)

The binary distribution of this product bundles these dependencies under the
following license:
JUnit 4.11
Eclipse JDT Core 3.1.1
--------------------------------------------------------------------------------
(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS
"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole
responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively “Losses”) arising from claims, lawsuits and other legal
actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

--------------------------------------------------------------------------------

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

--------------------------------------------------------------------------------

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0
--------------------------------------------------------------------------------

(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
"Licensor" means the individual or entity that offers the Work under the terms of this License.
"Original Author" means the individual or entity who created the Work.
"Work" means the copyrightable work of authorship offered under the terms of this License.
"You" means an individual or entity exercising rights under this License who has
not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

   - to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
   - to create and reproduce Derivative Works;
   - to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
   - to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

   - You may distribute, publicly display, publicly perform, or publicly digitally
perform the Work only under the terms of this License, and You must include a
copy of, or the Uniform Resource Identifier for, this License with every copy or
phonorecord of the Work You distribute, publicly display, publicly perform, or
publicly digitally perform. You may not offer or impose any terms on the Work
that alter or restrict the terms of this License or the recipients' exercise of
the rights granted hereunder. You may not sublicense the Work. You must keep
intact all notices that refer to this License and to the disclaimer of
warranties. You may not distribute, publicly display, publicly perform, or
publicly digitally perform the Work with any technological measures that control
access or use of the Work in a manner inconsistent with the terms of this
License Agreement. The above applies to the Work as incorporated in a Collective
Work, but this does not require the Collective Work apart from the Work itself
to be made subject to the terms of this License. If You create a Collective
Work, upon notice from any Licensor You must, to the extent practicable, remove
from the Collective Work any credit as required by clause 4(b), as requested. If
You create a Derivative Work, upon notice from any Licensor You must, to the
extent practicable, remove from the Derivative Work any credit as required by
clause 4(b), as requested.
If you distribute, publicly display, publicly perform, or publicly digitally
perform the Work or any Derivative Works or Collective Works, You must keep
intact all copyright notices for the Work and provide, reasonable to the medium
or means You are utilizing: (i) the name of the Original Author (or pseudonym,
if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor
designate another party or parties (e.g. a sponsor institute, publishing entity,
journal) for attribution in Licensor's copyright notice, terms of service or by
other reasonable means, the name of such party or parties; the title of the Work
if supplied; to the extent reasonably practicable, the Uniform Resource
Identifier, if any, that Licensor specifies to be associated with the Work,
unless such URI does not refer to the copyright notice or licensing information
for the Work; and in the case of a Derivative Work, a credit identifying the use
of the Work in the Derivative Work (e.g., "French translation of the Work by
Original Author," or "Screenplay based on original Work by Original Author").
Such credit may be implemented in any reasonable manner; provided, however, that
in the case of a Derivative Work or Collective Work, at a minimum such credit
will appear where any other comparable authorship credit appears and in a manner
at least as prominent as such other comparable authorship credit.
5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS
THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND
CONCERNING
THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT
LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY,
OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME
JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH
EXCLUSION MAY NOT APPLY TO YOU.
6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1
1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributors Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means
that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date
The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.
3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms
You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty
Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions
Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:

JDOM 1.1

/*--

OPEN SOURCE USED IN DNAC PLATFORM ARICA 2657
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgement may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information
on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an
Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the
h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in
which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for
attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URL, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS,
WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified
here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

--------------------------------------------------------------------------------

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.748 ret 0.1.15 0.1.15

1.748.1 Available under license:
Copyright (C) 2011 by Roly Fentanes

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.749 synchronous-promise 2.0.15 2.0.15

1.749.1 Available under license:
Copyright (c) 2016, Davyd McColl
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of synchronous-promise nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.750 lodash-find 4.6.0 4.6.0

1.750.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.751 traverse 0.6.6 0.6.6

1.751.1 Available under license:

Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:
http://www.opensource.org/licenses/mit-license.php

Copyright 2010 James Halliday (mail@substack.net)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.752 netty-buffer 4.1.59.Final

1.752.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/UnsafeHeapSwappedByteBuf.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/UnsafeDirectSwappedByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARNATIONS OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.
*/

Found in path(s):
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/AbstractPooledDerivedByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/SimpleLeakAwareCompositeByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/UnpooledUnsafeNoCleanerDirectByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/PooledDuplicatedByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/WrappedUnpooledUnsafeDirectByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/PooledSlicedByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/AdvancedLeakAwareCompositeByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/WrappedCompositeByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/io/netty/buffer/AbstractUnpooledSlicedByteBuf.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
    jar/META-INF/maven/io.netty/netty-buffer/pom.xml
No license file was found, but licenses were detected in source scan.

/*
  * Copyright 2012 The Netty Project

Open Source Used In DNAC Platform Arica 2670
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
jar/io/netty/buffer/UnpooledHeapByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/DuplicatedByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/UnpooledByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/SwappedByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/UnpooledByteBufAllocator.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/PooledByteBufAllocator.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/PooledDirectByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/ByteBufOutputStream.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/ByteBuf.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/PoolChunkList.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/PoolChunk.java  

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/search/package-info.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/LongPriorityQueue.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/SizeClassesMetric.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/LongLongHashMap.java  
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io/netty/buffer/SizeClasses.java
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/META-INF/native-image/io.netty/buffer/native-image.properties

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/PoolChunkListMetric.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/UnpooledDuplicatedByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/PooledUnsafeHeapByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/HeapByteBufUtil.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/UnsafeByteBufUtil.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/UnpooledUnsafeHeapByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/PoolSubpageMetric.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/UnpooledSlicedByteBuf.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-jar/io.netty/buffer/PoolArenaMetric.java
  * /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/AbstractUnsafeSwappedByteBuf.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/PoolChunkMetric.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2020 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* *
* https://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* */

Found in path(s):
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/AbstractSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/BitapSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/AhoCorasicSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/MultiSearchProcessor.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/MultiSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/KmpSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/SearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/AbstractMultiSearchProcessorFactory.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/SearchProcessor.java
* /opt/cola/permits/1145449558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
jar/io/netty/buffer/search/AbstractMultiSearchProcessorFactory.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2013 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/ReadOnlyUnsafeDirectByteBuf.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/AdvancedLeakAwareByteBuf.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/ByteBufProcessor.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/UnreleasableByteBuf.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/AbstractReferenceCountedByteBuf.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/DefaultByteBufHolder.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/PooledUnsafeDirectByteBuf.java
* /opt/cola/permits/114549558_1616156160.84/0/netty-buffer-4-1-59-final-sources-1-
  jar/io/netty/buffer/AbstractReferenceCountedByteBuf.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
1.753 lodash-pickby 4.6.0

1.753.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash
The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

**1.754 jackson-xc 2.10.4**

**1.754.1 Available under license:**

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.
This copy of Jackson JSON processor `jackson-module-jaxb-annotations` module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.755 opentype-js 1.1.0 1.1.0

1.755.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017 Frederik De Bleser

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
1.756 promise 7.3.1 7.3.1
1.756.1 Available under license :

Copyright (c) 2014 Forbes Lindesay

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.757 mongodb-driver-core 3.12.4
1.757.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz and Tim Peierls
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 *
 * Any republication or derived work distributed in source code form
 * must include this copyright and license notice.
 */

Found in path(s):
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/annotations/NotThreadSafe.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/annotations/Immutable.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/annotations/ThreadSafe.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original Work: MIT License, Copyright (c) [2015-2018] all contributors
 * https://github.com/marianobarrios/tls-channel
 */

Found in path(s):
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/WouldBlockException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/util/Util.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/TlsChannelBuilder.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannelGroup.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/util/TlsChannelCallbackException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/NeedsReadException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/impl/BufferHolder.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/impl/ExtendedAsynchronousByteChannel.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannel.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferUtil.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/ClientTlsChannel.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/tlschannel/BufferAllocator.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/event/ConnectionPoolClosedEvent.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/operation/QueryHelper.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/connection/ConnectionPoolSettings.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/internal/connection/DefaultServerConnection.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/connection/BufferProvider.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/client/model/geojson/codecs/MultiLineStringCodec.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/internal/connection/ClusterClockAdvancingSessionContext.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/internal/connection/DeleteProtocol.java
  */opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/event/ServerMonitorListener.java
Open Source Used In DNAC Platform Arica

* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/Projections.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ConnectionCheckedInEvent.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/geojson/CoordinateReferenceSystem.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/bulk/BulkWriteUpsert.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/UpdateManyModel.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/MongoInterruptedException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/InternalConnection.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/ServerCursor.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/bindings/AsyncSingleConnectionReadBinding.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/gridfs/codecs/GridFSFileCodecProvider.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/geojson/AbstractGeometryCodec.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/gridfs/codecs/GridFSFileCodecProvider.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/ClusterConnectionMode.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/geojson/NamedCoordinateReferenceSystem.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/gridfs/codecs/GridFSFileCodecProvider.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/AsynchronousSocketChannelStreamFactoryFactory.java
Open Source Used In DNAC Platform Arica

* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/vault/EncryptOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/diagnostics/logging/Logger.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/ReadPreference.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/management/ConnectionPoolStatistics.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ConnectionAddedEvent.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/CommandEventMulticaster.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/Server.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/WriteConcernResult.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/geojson/Polygon.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/FindAndDeleteOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/BulkWriteOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/Collation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/MapReduceWithInlineResultsOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/UserExistsOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/CollationCaseFirst.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/gridfs/model/GridFSDownloadOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/changestream/FullDocument.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/CommandListener.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/session/ClientSession.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/binding/AsyncConnectionSource.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ServerEventMulticaster.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/binding/AsyncWriteBinding.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/ReplyMessage.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/ClientSessionOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/DBRef.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/DeleteMessage.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ConnectionPoolOpenedEvent.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/CompressedMessage.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/dbRefCodec.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ServerMonitorListenerMulticaster.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/Authenticator.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/DeleteOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/WriteProtocol.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/Facet.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/SocketSettings.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/validator/CollectibleDocumentFieldNameValidator.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/PushOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/MongoInternalException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/MongoSocketOpenException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/IndexModel.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/binding/ReferenceCounted.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/AggregateResponseBatchCursor.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/ReplaceOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/bulk/BulkWriteError.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/event/ServerListener.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/SimpleExpression.java
Open Source Used In DNAC Platform Arica  2692

* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/selector/package-info.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/GetMoreMessage.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/client/model/FindOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/ClusterClock.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/session/SessionContext.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/connection/KillCursorProtocol.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/async/SingleResultCallback.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/client/model/FindOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/client/model/UnwindOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/client/model/changestream/ChangeStreamDocumentCodec.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/client/model/TextSearchOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/operation/UpdateUserOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/operation/ErrorCategory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/MongoGridFSEException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/connection/ClusterType.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/MongoServerException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/internal/event/ClusterListenerMulticaster.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-jar/com/mongodb/connection/netty/NettyStreamFactory.java
Open Source Used In DNAC Platform Articles  2696

* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/InternalConnectionFactory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/BaseCluster.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/DBObjectCodec.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/event/ServerClosedEvent.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/BsonWriterHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/connection/ClusterDescription.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/client/model/InsertManyOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/DefaultConnectionPool.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/ConnectionFactory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/client/model/EstimatedDocumentCountOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/client/model/CountOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/client/model/GraphLookupOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/ConnectionFactory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/DBObject.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/management/jmx/JMXMBeanServer.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/connection/netty/NettyStreamFactoryFactory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/client/model/geojson/codecs/GeometryDecoderHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/MongoWaitQueueFullException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/GetMoreProtocol.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/session/BaseClientSessionImpl.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/internal/connection/UsageTrackingInternalConnection.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/operation/MixedBulkWriteOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
jar/com/mongodb/MongoBulkWriteException.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/management/MBeanServer.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/ConnectionString.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/geojson/MultiLineString.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/AsynchronousChannelStream.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/TagSet.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/client/model/InsertOneOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/BaseFindAndModifyOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/MapReduceInlineResultsAsyncCursor.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/FindAndReplaceOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/RenameCollectionOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/TlsChannelStreamFactoryFactory.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/LegacyProtocol.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/GroupOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/FindAndModifyHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/connection/ConnectionId.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
/* Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/operation/ServerVersionHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/CommitTransactionOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/TransactionOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/TransactionOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/ReadConcernAwareNoOpSessionContext.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/ReadConcernHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/AbortTransactionOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/OperationReadConcernHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/lang/Nullable.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/lang/NonNullApi.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/lang/NonNull.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/lang/package-info.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/AbortTransactionOperation.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/lang/NonNull.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/internal/connection/ReadConcernHelper.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/TransactionOptions.java
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1.jar/com/mongodb/operation/OperationReadConcernHelper.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2008-present MongoDB, Inc.
* Copyright 2012 The Netty Project
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /
/*
* Copyright 2008-present MongoDB, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original Work: MIT License, Copyright (c) [2015-2018] all contributors
* https://github.com/marianobbarrios/tls-channel
*/
/**
* Writes a sequence of bytesProduced to this channel from a subsequence of the
* given buffers.
*
* @param srcs   The buffers from which bytesProduced are to be retrieved
* @param offset The offset within the buffer array of the first buffer from
*                which bytesProduced are to be retrieved; must be non-negative and no
*                larger than <tt>srcs.length</tt>
* @param length The maximum number of buffers to be accessed; must be non-negative and no larger than <tt>srcs.length</tt>
* @return The number of bytesProduced written, contrary to the behavior specified
* in { @link ByteChannel}, this method never returns 0, but throws
* { @link WouldBlockException} and
* @throws IndexOutOfBoundsException If the preconditions on the <tt>offset</tt> and
*/
/*
 * length parameters do not hold
 * @throws WouldBlockException if the channel is in non-blocking mode and the IO operation
 * cannot be completed immediately
 * @throws NeedsTaskException if the channel is not configured to run tasks automatically
 * and a task needs to be executed to complete the operation
 * @throws SSLException if the {@link SSLEngine} throws a SSLException
 * @throws IOException if the underlying channel throws an IOException
 */

/**
 * Reads a sequence of bytesProduced from this channel into a subsequence of the
 * given buffers.
 *
 * <p>
 * See {@link ScatteringByteChannel#read(ByteBuffer[], int, int)} for more
 * details of the meaning of this signature.
 *
 * <p>
 * This method behaves slightly different than the interface specification,
 * with respect to non-blocking responses, see {@link #read(ByteBuffer)} for
 * more details.
 *
 * @param dsts   The buffers into which bytesProduced are to be transferred
 * @param offset The offset within the buffer array of the first buffer into
 * which bytesProduced are to be transferred; must be non-negative and no
 * larger than <tt>dsts.length</tt>
 * @param length The maximum number of buffers to be accessed; must be
 * non-negative and no larger than <tt>dsts.length</tt>
 * &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&n...
* Copyright 2008-present MongoDB, Inc.
* Copyright 2017 Tom Bentley
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/internal/authentication/SaslPrep.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008-present MongoDB, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/

/**
* Returns an array containing all of the elements in this deque, in
* proper sequence (from first to last element).
* <p>The returned array will be "safe" in that no references to it are
* maintained by this deque. (In other words, this method must allocate
* a new array). The caller is thus free to modify the returned array.
This method acts as bridge between array-based and collection-based APIs.

@return an array containing all of the elements in this deque
*/

/*
* Copyright 2008-present MongoDB, Inc.
* Copyright (c) 2008-2014 Atlassian Pty Ltd
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/*
* Copyright 2008-present MongoDB, Inc.
* Copyright 2010 The Guava Authors
* Copyright 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*/
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136447791_1613767063.08/0/mongodb-driver-core-3-12-4-sources-1-
  jar/com/mongodb/annotations/Beta.java

1.758 babel-plugin-transform-es2015-literals
6.22.0

1.758.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.759 ip-address 5.3.1

1.759.1 Available under license:

No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/IPAddressResources.properties

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2020 Sean C Foley
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * or at
 * https://github.com/seancfoley/IPAddress/blob/master/LICENSE
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/IPAddressSection.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 Sean C Foley
*
* Licensed under the Apache License, Version 2.0 (the "License"); 
* you may not use this file except in compliance with the License. 
* You may obtain a copy of the License at 
* 
* http://www.apache.org/licenses/LICENSE-2.0 
* or at 
* https://github.com/seancfoley/IPAddress/blob/master/LICENSE 
* 
* Unless required by applicable law or agreed to in writing, software 
* distributed under the License is distributed on an "AS IS" BASIS, 
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
* See the License for the specific language governing permissions and 
* limitations under the License. 
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/HostName.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/validate/ParsedHost.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/string/package-info.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/standard/IPAddressJoinedSegments.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/standard/package-info.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/mac/MACAddressSegment.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/ipv4/IPv4AddressSection.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/standard/IPAddressJoinedSegments.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/AddressPositionException.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/standard/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 Sean C Foley
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     http://www.apache.org/licenses/LICENSE-2.0
 *     or at
 *     https://github.com/seancfoley/IPAddress/blob/master/LICENSE
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/IPv4AddressTrie.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/mac/MACAddressTrie.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/mac/MACAddressAssociativeTrie.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/ipv6/IPv6AddressAssociativeTrie.java
* /opt/ws_local/PERMITS_SQL/1077201543_1596778639.63/0/ipaddress-5-3-1-sources-jar/inet/ipaddr/format/util/Partition.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 Sean C Foley
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * or at
 * https://github.com/seancfoley/IPAddress/blob/master/LICENSE
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
1.760 hadoop 1.2.1

1.760.1 Available under license:

JZlib 0.0.* were released under the GNU LGPL license. Later, we have switched over to a BSD-style license.

------------------------------------------------------------------
Copyright (c) 2000,2001,2002,2003 ymnk, JCraft, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT,
INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA,
OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.

Copyright 2000-2007 Sun Microsystems, Inc. All rights reserved.

The contents of this file are subject to the terms of either the GNU
General Public License Version 2 only ("GPL") or the Common Development
and Distribution License ("CDDL") (collectively, the "License"). You
may not use this file except in compliance with the License. You can obtain
a copy of the License at https://glassfish.dev.java.net/public/CDDL+GPL.html
or glassfish/bootstrap/legal/LICENSE.txt. See the License for the specific
language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each
file and include the License file at glassfish/bootstrap/legal/LICENSE.txt.
Sun designates this particular file as subject to the "Classpath" exception
as provided by Sun in the GPL Version 2 section of the License file that
accompanied this code. If applicable, add the following below the License
Header, with the fields enclosed by brackets [ ] replaced by your own
identifying information: "Portions Copyrighted [year]
[name of copyright owner]"

Contributor(s):

If you wish your version of this file to be governed by only the CDDL or
only the GPL Version 2, indicate your decision by adding "[Contributor]
elects to include this software in this distribution under the [CDDL or GPL
Version 2] license." If you don't indicate a single choice of license, a
recipient has the option to distribute your version of this file under
either the CDDL, the GPL Version 2 or to extend the choice of license to
its licensees as provided above. However, if you add GPL Version 2 code
and therefore, elected the GPL Version 2 license, then the option applies
only if the new code is made subject to such option by the copyright
holder.

-->

/* Copyright (c) 1995-2000, The Hypersonic SQL Group. */
* All rights reserved.

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.

* Neither the name of the Hypersonic SQL Group nor the names of its
* contributors may be used to endorse or promote products derived from this
* software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE HYPersonic SQL GROUP,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* This software consists of voluntary contributions made by many individuals
* on behalf of the Hypersonic SQL Group.

*
*
*
* For work added by the HSQL Development Group:

*

* Copyright (c) 2001-2004, The HSQL Development Group
*
* All rights reserved.
*
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
*
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*
* Neither the name of the HSQL Development Group nor the names of its
* contributors may be used to endorse or promote products derived from this

* software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

* ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG,

* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,

* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,

* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND

* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright
owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to
any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by
brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and
description of purpose be included on the same "printed page" as the copyright notice for easier identification within
third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you
may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS
OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
limitations under the License.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses

grounded to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You

meet the following conditions:

(a) You must give any other recipients of the Work or

    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices

    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

    that You distribute, all copyright, patent, trademark, and

    attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and
limitations under the License.

/* Copyright (c) 1995-2000, The Hypersonic SQL Group.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 * Neither the name of the Hypersonic SQL Group nor the names of its
 * contributors may be used to endorse or promote products derived from this
 * software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
 * OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * This software consists of voluntary contributions made by many individuals
 * on behalf of the Hypersonic SQL Group.
 *
 *
 * For work added by the HSQL Development Group:
 *
 * Copyright (c) 2001-2004, The HSQL Development Group
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, * this list of conditions and the following disclaimer in the documentation * and/or other materials provided with the distribution.
* Neither the name of the HSQL Development Group nor the names of its * contributors may be used to endorse or promote products derived from this * software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE * ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, * OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following
* conditions are met:
  * Redistributions of source code must retain the above copyright
  * notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright
  * notice, this list of conditions and the following disclaimer in
  * the documentation and/or other materials provided with the distribution.
  * Neither the name of the University Catholique de Louvain - UCL
  * nor the names of its contributors may be used to endorse or
  * promote products derived from this software without specific prior
  * written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
  * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
  * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
  * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
  * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
  * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
  * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
  * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
  * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
  * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
  * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
  * POSSIBILITY OF SUCH DAMAGE.
  */
/**
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a ”NOTICE” text file as part of its
distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright
notices and license terms. Your use of the source code for these
subcomponents is subject to the terms and conditions of the following
licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
* - Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* - Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the distribution.

* - Neither the name of the University Catholique de Louvain - UCL

* nor the names of its contributors may be used to endorse or

* promote products derived from this software without specific prior

* written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT

* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS

* FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE

* COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,

* BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER

* CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN

* ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE

* POSSIBILITY OF SUCH DAMAGE.

*/

Apache MINA
Copyright 2007 The Apache Software Foundation.
This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Please refer to each LICENSE.<component>.txt file for the license terms of the components that Apache MINA depends on.

Message logging is provided by the SLF4J library package, which is open source software, written by Ceki Glč, and copyright by SLF4J.ORG and QOS.ch. The original software is available from

http://www.slf4j.org/

Data compression support is provided by the JZLib library package, which is open source software, written by JCraft, and copyright by JCraft. The original software is available from

http://www.jcraft.com/jzlib/

Spring framework is provided by the Spring framework library package, which is open source software, written by Rod Johnson et al, and copyright by Springframework.org. The original software is available from

http://www.springframework.org/

OGNL is provided by the OGNL library package, which is open source software, written by Drew Davidson and Luke Blanshard. The original software is available from

http://www.ognl.org/

Thrift Software License

Copyright (c) 2006- Facebook, Inc.

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor
(if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by
the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR
MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under
copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications
and/or translated into another language. (Hereinafter, translation is included without limitation in the term
"modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its
scope. The act of running the Program is not restricted, and the output from the Program is covered only if its
contents constitute a work based on the Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium,
provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any
change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started
running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-
free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY
COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL,
SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY
TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER
PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to
achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to
most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later
version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the
implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and
you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.
Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright
disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words
"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Thrift Software License
Copyright (c) 2006- Facebook, Inc.

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

OGNL is the creation of Luke Blanshard and Drew Davidson.

Copyright 1997-2003, Drew Davidson and Luke Blanshard. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Drew Davidson nor the names of its contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<!--
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
-->  

1.761 glib-networking 2.48.2 1~ubuntu16.04.1

1.761.1 Available under license:

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This
license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don’t assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data
prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no
charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6.
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies
It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and
conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice

That's all there is to it!

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy,
and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Files: *
Copyright: 2009-2010 Red Hat, Inc.
    2010 Collabora, Ltd.
License: LGPL-2+

Files: po/bg.po
    po/eo.po
    po/sl.po
    po/sv.po
    po/th.po
    po/tr.po
Copyright: 2011-2013 Free Software Foundation
License: LGPL-2+

Files: po/fi.po
Copyright: 2011 Tommi Vainikainen
License: LGPL-2+

Files: po/te.po
Copyright: 2011-2012 Swecha Telugu Localisation team
License: LGPL-2+

Files: po/vi.po
Copyright: 2011 Free Software Foundation
License: LGPL-2+

Files: tls/tests/mock-pkcs11.*
Copyright: 2010 Stefan Walter
    2011 Collabora Ltd.
License: LGPL-2.1+

License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.


This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library. If not, see <http://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in "/usr/share/common-licenses/LGPL-2".

License: LGPL-2.1+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library. If not, see <http://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in "/usr/share/common-licenses/LGPL-2.1".

1.762 commons-codec 1.14
1.762.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
 incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Codec
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

1.763 source-map 0.6.1 0.6.1
1.763.1 Available under license :

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project
  contributors may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.764 find-up 4.1.0 4.1.0

1.764.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.765 locate-path 5.0.0

1.765.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.766 bunyan-format 0.2.1 0.2.1

1.766.1 Available under license:

Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.767 es6-promise 3.3.1 3.3.1

1.767.1 Available under license:

Copyright (c) 2014 Yehuda Katz, Tom Dale, Stefan Penner and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.768 is-string 1.0.7
1.768.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.769 json 1.8
1.769.1 Available under license :
No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/META-INF/maven/com.tdunning/json/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Android Open Source Project
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSON.java
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSONTokener.java
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSONObject.java
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSONStringer.java
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSONArray.java
* /opt/cola/permits/1258286420_1643052502.2/0/json-1-8-sources-jar/org/json/JSONException.java

1.770 unset-value 1.0.0 1.0.0
1.770.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.771 calcite-core 1.26.0

1.771.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Calcite
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.
The web site includes files generated by Jekyll.

1.772 lcid 1.0.0 1.0.0
1.772.1 Available under license :
   The MIT License (MIT)

   Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
   all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
   THE SOFTWARE.

1.773 debug 4.3.4
1.773.1 Available under license :
   (The MIT License)

   Copyright (c) 2014-2017 TJ Holowaychuk <tj@vision-media.ca>
   Copyright (c) 2018-2021 Josh Junon

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software
   and associated documentation files (the 'Software'), to deal in the Software without restriction,
   including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
   and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
   subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial
   portions of the Software.

   THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
   INCLUDING BUT NOT
1.774 jcip-annotation 1.0

1.774.1 Available under license:

No license file was found, but licenses were detected in source scan.

Class, field, and method level annotations for describing thread-safety policies.

Three class-level annotations describe the *intended* thread-safety promises of a class: `@Immutable`, `@ThreadSafe`, and `@NotThreadSafe`. `@Immutable` means that the class is immutable, and implies `@ThreadSafe`. `@NotThreadSafe` is optional; if a class is not annotated as thread-safe, it should be presumed not to be thread-safe, but if you want to make it extra clear, use `@NotThreadSafe`.

These annotations are relatively unintrusive and are beneficial to both users and maintainers. Users can see immediately whether a class is thread-safe, and maintainers can see immediately whether thread-safety guarantees must be preserved. Annotations are also useful to a third constituency: tools. Static code-analysis tools may be able to verify that the code complies with the contract indicated by the annotation, such as verifying that a class annotated with `@Immutable` actually is immutable.

Field and method annotations

The class-level annotations above are part of the public documentation for the class. Other aspects of a class's thread-safety strategy are entirely for maintainers and are not part of its public documentation.
Classes that use locking should document which state variables are guarded with which locks, and which locks are used to guard those variables. A common source of inadvertent non-thread-safety is when a thread-safe class consistently uses locking to guard its state, but is later modified to add either new state variables that are not adequately guarded by locking, or new methods that do not use locking properly to guard the existing state variables. Documenting which variables are guarded by which locks can help prevent both types of omissions.

The `<code>@GuardedBy(lock)</code>` annotation documents that a field or method should be accessed only with a specific lock held.

<h2>Copyright and license</h2>

This software is copyright (c) 2005 Brian Goetz and Tim Peierls and is released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5). The official home for this software is http://www.jcip.net. Any republication or derived work distributed in source code form must include the copyright and license notice.

Found in path(s):
* /opt/cola/permits/1135913176_1613629761.33/0/jcip-annotations-1-0-sources-2-jar/net/jcip/annotations/package.html

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz and Tim Peierls
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 *
 * Any republication or derived work distributed in source code form
 * must include this copyright and license notice.
 */
1.775 react-highlight-words 0.16.0 0.16.0

1.775.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Treasure Data

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.776 @babel/helper-plugin-utils 7.13.0

1.776.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.777 prop-types 15.6.2 15.6.2

1.777.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.778 parquet-format-structures 1.12.0
1.778.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/event/TypedConsumer.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/InterningProtocol.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/EventBasedThriftReader.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/BlockCipher.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/FieldConsumer.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/Util.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/Event/Consumers.java
* /opt/cola/permits/1202446978_1631568294.17/0/parquet-format-structures-1-12-0-sources-
  jar/org/apache/parquet/format/LogicalTypes.java

No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 */
1.779 babel-plugin-syntax-top-level-await

7.14.5

1.779.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.780 annotations 2.0.1

1.780.1 Available under license:

Licenses
========

Good news! You may use Kdyby Framework under the terms of either the New BSD License or the GNU General Public License (GPL) version 2 or 3.

The BSD License is recommended for most projects. It is easy to understand and it places almost no restrictions on what you can do with the framework. If the GPL fits better to your project, you can use the framework under this license.

You don't have to notify anyone which license you are using. You can freely use Kdyby Framework in commercial projects as long as the copyright header remains intact.

New BSD License

Copyright (c) 2008 Filip Prochzka (http://filip-prochazka.com)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of "Kdyby Framework" nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort.
(including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

GNU General Public License

--------------------------

GPL licenses are very very long, so instead of including them here we offer you URLs with full text:

- [GPL version 2](http://www.gnu.org/licenses/gpl-2.0.html)
- [GPL version 3](http://www.gnu.org/licenses/gpl-3.0.html)

### 1.781 sanitize-html-react 1.13.0 1.13.0

#### 1.781.1 Available under license:

**Copyright (c) 2013, 2014, 2015 P'unk Avenue LLC**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.782 statuses 1.2.1 1.2.1

#### 1.782.1 Available under license:

**The MIT License (MIT)**

**Copyright (c) 2014 Jonathan Ong me@jongleberry.com**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.783 babel-plugin-transform-es2015-classes
6.24.1 6.24.1
1.783.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.784 async 1.0.0 1.0.0
1.784.1 Available under license :

Copyright (c) 2010-2014 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.785 agent-base 4.3.0 4.3.0
1.785.1 Available under license :

Copyright (c) 2013 Nathan Rajlich &lt;nathan@tootallnate.net&gt;

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.786 camelcase 2.1.1 2.1.1
**1.786.1 Available under license:**

The MIT License (MIT)

Copyright (c) Sindre Sorhus &lt;sindresorhus@gmail.com&gt; (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

**1.787 pixman 0.33.6 1**

**1.787.1 Available under license:**

The following is the MIT license, agreed upon by most contributors.
Copyright holders of new code should use this license statement where
possible. They may also add themselves to the list below.

/*
 *  Copyright 1987, 1988, 1989, 1998  The Open Group
 *  Copyright 1987, 1988, 1989 Digital Equipment Corporation
 *  Copyright 1999, 2004, 2008 Keith Packard
 *  Copyright 2000 SuSE, Inc.
 *  Copyright 2000 Keith Packard, member of The XFree86 Project, Inc.
 *  Copyright 2004 Nicholas Miell
 *  Copyright 2005 Lars Knoll & Zack Rusin, Trolltech
 *  Copyright 2005 Trolltech AS
 *  Copyright 2007 Luca Barbato
 *  Copyright 2008 Aaron Plattner, NVIDIA Corporation
 *  Copyright 2008 Rodrigo Kumpera
 *  Copyright 2008 Andr Tupinamb
 *  Copyright 2008 Mozilla Corporation
 *  Copyright 2008 Frederic Plourde
 *  Copyright 2009, Oracle and/or its affiliates. All rights reserved.
1.788 drill-iceberg-metastore 1.19.0

1.788.1 Available under license:

Drill : Metastore : Iceberg
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.789 isexe 2.0.0 2.0.0
1.789.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.790 is-number 7.0.0

1.790.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.791 unicode-trie 0.3.1 0.3.1
1.791.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.792 commander 4.1.1

1.792.1 Available under license:

No license file was found, but licenses were detected in source scan.

# Permission is hereby granted, free of charge, to any person obtaining
# a copy of this software and associated documentation files (the
# "Software"), to deal in the Software without restriction, including
# without limitation the rights to use, copy, modify, merge, publish,
# distribute, sublicense, and/or sell copies of the Software, and to
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.

Found in path(s):
* /opt/cola/permits/1163925133_1620933582.81/0/commander-4-1-1-gem/data-tar-gz/lib/commander.rb
No license file was found, but licenses were detected in source scan.

= Commander

The complete solution for Ruby command-line executables.
Commander bridges the gap between other terminal related libraries
you know and love (OptionParser, HighLine), while providing many new
features, and an elegant API.

== Features

* Easier than baking cookies
* Parses options using OptionParser
* Auto-populates struct with options (no more { |v| options[:recursive] = v } )
* Auto-generates help documentation via pluggable help formatters
* Optional default command when none is present
* Global / Command level options
* Packaged with two help formatters (Terminal, TerminalCompact)
* Imports the highline gem for interacting with the terminal
* Adds additional user interaction functionality
* Highly customizable progress bar with intuitive, simple usage
* Multi-word command name support such as 'drupal module install MOD', rather than 'drupal module_install MOD'
* Sexy paging for long bodies of text
* Support for MacOS text-to-speech
* Command aliasing (very powerful, as both switches and arguments can be used)
* Growl notification support for MacOS
* Use the 'commander' executable to initialize a commander driven program

== Installation

$ gem install commander

== Quick Start

To generate a quick template for a commander app, run:

$ commander init yourfile.rb

== Example

For more option examples view the Commander::Command#option method. Also an important feature to note is that action may be a class to instantiate, as well as an object, specifying a method to call, so view the RDoc for more information.

require 'rubygems'
require 'commander/import'

# :name is optional, otherwise uses the basename of this executable
program :name, 'Foo Bar'
program :version, '1.0.0'
program :description, 'Stupid command that prints foo or bar.'

command :foo do |
  c.syntax = 'foobar foo'
  c.description = 'Displays foo'
  c.action do |args, options|
    say 'foo'
  end
end
command :bar do |c|
  c.syntax = 'foobar bar [options]' 
  c.description = 'Display bar with optional prefix and suffix'
  c.option '--prefix STRING', String, 'Adds a prefix to bar'
  c.option '--suffix STRING', String, 'Adds a suffix to bar'
  c.action do |args, options|
    options.default :prefix => '(', :suffix => ')' 
    say "#{options.prefix}bar#{options.suffix}" 
  end
end

$ foobar bar
# => (bar)

$ foobar bar --suffix '{' --prefix '}'
# => {bar}

== HighLine

As mentioned above the highline gem is imported into 'global scope', below are some quick examples for how to utilize highline in your command(s):

# Ask for password masked with '*' character
ask("Password: ") { |q| q.echo = "*" }

# Ask for password
ask("Password: ") { |q| q.echo = false }

# Ask if the user agrees (yes or no)
agree("Do something?")

# Asks on a single line (note the space after ":")
ask("Name: ")

# Asks with new line after "Description:"
ask("Description:" )

# Calls Date#parse to parse the date string passed
ask("Birthday? ", Date)

# Ensures Integer is within the range specified
ask("Age? ", Integer) { |q| q.in = 0..105 }

# Asks for a list of strings, converts to array
ask("Fav colors?", Array)

== HighLine & Interaction Additions
In addition to highline's fantastic choice of methods, commander adds the following methods to simplify common tasks:

# Ask for password
password

# Ask for password with specific message and mask character
password "Enter your password please:", '-'

# Ask for CLASS, which may be any valid class responding to #parse. Date, Time, Array, etc
names = ask_for_array 'Names: '
bday = ask_for_date 'Birthday?: '

# Simple progress bar (Commander::UI::ProgressBar)
uris = %w[
  http://vision-media.ca
  http://google.com
  http://yahoo.com
]
progress uris do |uri|
  res = open uri
  # Do something with response
end

# 'Log' action to stdout
log "create", "path/to/file.rb"

# Enable paging of output after this point
enable_paging

# Ask editor for input (EDITOR environment variable or whichever is available: TextMate, vim, vi, emacs, nano, pico)
ask_editor

# Ask editor, supplying initial text
ask_editor 'previous data to update'

# Ask editor, preferring a specific editor
ask_editor 'previous data', 'vim'

# Choose from an array of elements
choice = choose("Favorite language?", :ruby, :perl, :js)

# Alter IO for the duration of the block
io new_input, new_output do
  new_input_contents = $stdin.read
  puts new_input_contents # outputs to new_output stream
# Speech synthesis
speak 'What is your favorite food?'
food = ask 'favorite food?: '
speak "Wow, I like #{food} too. We have so much in common."

# Execute arbitrary applescript
applescript 'foo'

# Converse with speech recognition server
case converse 'What is the best food?', :cookies => 'Cookies', :unknown => 'Nothing'
when :cookies
  speak 'o.m.g. you are awesome!'
else
  case converse 'That is lame, shall I convince you cookies are the best?', :yes => 'Ok', :no => 'No', :maybe =>
  'Maybe another time'
when :yes
  speak 'Well you see, cookies are just fantastic, they melt in your mouth.'
else
  speak 'Ok then, bye.'
end
end

== Growl Notifications

Commander provides methods for displaying Growl notifications. To use these methods you need to install http://github.com/visionmedia/growl which utilizes the growlnotify[http://growl.info/extras.php#growlnotify] executable. Note that growl is auto-imported by Commander when available, no need to require.

# Display a generic Growl notification
notify 'Something happened'

# Display an 'info' status notification
notify_info 'You have #{emails.length} new email(s)'

# Display an 'ok' status notification
notify_ok 'Gems updated'

# Display a 'warning' status notification
notify_warning '1 gem failed installation'

# Display an 'error' status notification
notify_error "Gem #{name} failed"

growlnotify can also be installed via homebrew[http://mxcl.github.com/homebrew/]:

end
# $stdin / $stdout reset back to original streams
$ brew install growlnotify

== Commander Goodies

=== Option Defaults

The options struct passed to #action provides a #default method, allowing you to set defaults in a clean manner for options which have not been set.

```ruby
command :foo do |c|
  c.option '--interval SECONDS', Integer, 'Interval in seconds'
  c.option '--timeout SECONDS', Integer, 'Timeout in seconds'
  c.action do [args, options]
    options.default |
      :interval => 2,
      :timeout  => 60
    end
  end
end
```

=== Command Aliasing

Aliases can be created using the #alias_command method like below:

```ruby
command :'install gem' do |c|
  c.action { puts 'foo' }
end
alias_command :'gem install', :'install gem'
```

Or more complicated aliases can be made, passing any arguments as if it was invoked via the command line:

```ruby
command :'install gem' do |c|
  c.syntax = 'install gem <name> [options]'
  c.option '--dest DIR', String, 'Destination directory'
  c.action { |args, options| puts "installing #{args.first} to #{options.dest}" }
end
alias_command :update, :'install gem', 'rubygems', '--dest', 'some_path'
```

$ foo update
# => installing rubygems to some_path

=== Command Defaults

Although working with a command executable framework provides many benefits over a single command implementation, sometimes you still want the ability to create a terse syntax for your command. With that in mind we may use #default_command to help with this. Considering
our previous ‘install gem’ example:

default_command :update

$ foo
# => installing rubygems to some_path

Keeping in mind that commander searches for the longest possible match when considering a command, so if you were to pass arguments to foo like below, expecting them to be passed to :update, this would be incorrect, and would end up calling ‘install gem’, so be careful that the users do not need to use command names within the arguments.

$ foo install gem
# => installing to

=== Additional Global Help

Arbitrary help can be added using the following #program symbol:

program :help, 'Author', 'TJ Holowaychuk <tj@vision-media.ca>'

Which will output the rest of the help doc, along with:

AUTHOR:

TJ Holowaychuk <tj@vision-media.ca>

=== Global Options

Although most switches will be at the command level, several are available by default at the global level, such as --version, and --help. Using #global_option you can add additional global options:

```
global_option('-c', '--config FILE', 'Load config data for your commands to use') { |file| ... }
```

This method accepts the same syntax as Commander::Command#option so check it out for documentation.

All global options regardless of providing a block are accessible at the command level. This means that instead of the following:

```
global_option('--verbose') { $verbose = true }
... c.action do [args, options]
  say 'foo' if $verbose
  ...
```

You may:
global_option '--verbose'
...
c.action do |args, options|
  say 'foo' if options.verbose
...

=== Formatters

Two core formatters are currently available, the default Terminal formatter as well as TerminalCompact. To utilize a different formatter simply use :help_formatter like below:

program :help_formatter, Commander::HelpFormatter::TerminalCompact

Or utilize the help formatter aliases:

program :help_formatter, :compact

This abstraction could be utilized to generate HTML documentation for your executable.

== Tips

When adding a global or command option, OptionParser implicitly adds a small switch even when not explicitly created, for example -c will be the same as --config in both examples, however '-c' will only appear in the documentation when explicitly assigning it.

global_option '-c', '--config FILE'
global_option '--config FILE'

== ASCII Tables

For feature rich ASCII tables for your terminal app check out visionmedia's terminal-table gem at http://github.com/visionmedia/terminal-table

+----------+-------+----+--------+-----------------------+
| Terminal | Table | Is | Wicked | Awesome               |
+----------+-------+----+--------+-----------------------+
|          |       |    |        | get it while its hot! |
+----------+-------+----+--------+-----------------------+

== Running Specifications

$ rake spec

OR

$ spec --color spec
== Contrib

Feel free to fork and request a pull, or submit a ticket
http://github.com/visionmedia/commander/issues

== License

(The MIT License)

Copyright (c) 2008-2009 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Found in path(s):
* /opt/cola/permits/1163925133_1620933582.81/0/commander-4-1-1-gem/data-tar-gz/README.rdoc

1.793 nomnom 1.8.1 1.8.1

1.793.1 Available under license:

Copyright (c) 2010 Heather Arthur

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.794 asn1 0.2.4 0.2.4

1.794.1 Available under license:
Copyright (c) 2011 Mark Cavage, All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
1.795 component-bind 1.0.0 1.0.0

1.795.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.796 @babel/compat-data 7.16.4

1.796.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.797 httpcore-nio 4.4.10

1.797.1 Available under license:

Apache HttpCore NIO
Copyright 2005-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.798 @babel/runtime 7.0.0-beta.44 7.0.0-beta.44
1.798.1 Available under license:
MIT License

Copyright (c) 2014-2018 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.799 invariant 2.2.4
1.799.1 Available under license:
MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.800 hk2-locator 2.2.0-b21

1.800.1 Available under license:
Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in '* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

1.801 json-stringify-safe 5.0.1 5.0.1

1.801.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.802 babel-plugin-transform-es2015-modules-amd 6.24.1

1.802.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE 
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION 
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright 
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright 
  notice, this list of conditions and the following disclaimer in the 
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" 
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY 
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES 
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; 
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND 
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF 
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all 
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.803 jansi 1.18
1.803.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherefor such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.804 is-data-descriptor 1.0.0

1.804.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.805 groovy-datetime 3.0.8

1.805.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1218311999_1634926077.36/0/groovy-datetime-3-0-8-sources-jar/org/apache/groovy/datetime/extensions/DateTimeStaticExtensions.java
* /opt/cola/permits/1218311999_1634926077.36/0/groovy-datetime-3-0-8-sources-jar/org/apache/groovy/datetime/extensions/DateTimeExtensions.java
1.806 kerb-admin 1.0.0
1.806.1 Available under license:

Kerby-kerb Admin
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersed or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.807 http2-client 1.3.3 1.3.3
1.807.1 Available under license :
    MIT License

Copyright (c) 2019 Eyal.D <hisco@googlegroups.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
doing associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
1.808 jbig-kit 2.1-3.1
1.808.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free

Open Source Used In DNAC Platform Arica  2852
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law; that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Open Source Used In DNAC Platform Arica  2858
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: jbigkit
Source: http://www.cl.cam.ac.uk/~mgk25/jbigkit/

Files: *
Copyright: 1995-2008 Dr. Marcus Kuhn <mgk25@cl.cam.ac.uk>
License: GPL-2+
This package is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>

On Debian systems, the complete text of the GNU General Public License version 2 can be found in "/usr/share/common-licenses/GPL-2".

Files: debian/*
Copyright: 2011 Michael van der Kolff <mvanderkolff@gmail.com>
License: GPL-2+
This package is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>

On Debian systems, the complete text of the GNU General Public License version 2 can be found in "/usr/share/common-licenses/GPL-2".
1.809 hbase-replication 2.4.2
1.809.1 Available under license:

Apache HBase - Replication
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.810 passport-http-bearer 1.0.1 1.0.1
1.810.1 Available under license :
(The MIT License)

Copyright (c) 2011-2013 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.811 hk2-locator 2.5.0-b32

1.811.1 Available under license :
CDDL + GPLv2 with classpath exception
https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
A business-friendly OSS license

1.812 prefixmap 2.0

1.812.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Yet Another UserAgent Analyzer
 * Copyright (C) 2013-2020 Niels Basjes
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/prefixmap/src/test/java/nl/basjes/collections/prefixmap/serialization/AbstractSerializeWithKryo.java
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/serialization/test-base/src/test/java/nl/example/serialization/AbstractSerializationTest.java
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/serialization/test-base/src/test/java/nl/example/serialization/SerializeStringPrefixMapWithJava.java
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/serialization/test-base/src/test/java/nl/example/serialization/AbstractSerializeWithKryo.java
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/serialization/test-base/src/test/java/nl/example/serialization/SerializeASCIIPrefixMapWithKryoRegistered.java
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-prefixmap-e34e4cb/serialization/test-base/src/test/java/nl/example/serialization/SerializeStringPrefixMapWithKryo.java
No license file was found, but licenses were detected in source scan.

---

*/

* Copyright (C) 2018-2020 Niels Basjes

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
This is intended as an overview of the major changes

v2.0

- The PrefixMap now implements the Map interface which increases the number of use cases.
- Support for the default Java Serialization and Kryo 4 and Kryo 5

v1.2

- Converted this into a Java 9+ module that can still be used as a Java 8 jar dependency.

v1.1

- Support for serialization using Kryo
v1.0
===
- Initial release

License
=======
Copyright (C) 2018-2020 Niels Basjes

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-
  prefixmap-e34e4cb/CHANGELOG.md

No license file was found, but licenses were detected in source scan.

#
# Copyright (C) 2013-2018 Niels Basjes
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# #
# # https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an AS IS BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
#
root = true

[*]
charset = utf-8
end_of_line = lf
insert_final_newline = true
trim_trailing_whitespace = true
PrefixMap: A Map type datastructure that uses prefixes instead of keys

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,

No license file was found, but licenses were detected in source scan.

PrefixMap: A Map type datastructure that uses prefixes instead of keys

[![Travis Build status](https://api.travis-ci.org/nielsbasjes/prefixmap.png?branch=master)][https://travis-ci.org/nielsbasjes/prefixmap]


[![License](https://img.shields.io/license-apache-blue.svg)](https://www.apache.org/licenses/LICENSE-2.0.html)
This is a java library that implements a datastructure I consider to be missing in the java landscape.

The datastructure and its usages
================================
In several situations I have a set of values that are each associated with a different prefix of a String.

The example where I found the biggest need for this was when analyzing the device tags of mobile devices.

- A "GT-I8190N" is a "Samsung" device because it starts with "GT-".
- A "RM-1092" is a "Nokia" device because it starts with "RM-".

So I needed a lookup structure that maps these prefixes to the desired brands.

"GT-" : "Samsung"
"RM-" : "Nokia"

With this code you can now do this type of lookup in a very fast way:

// Parameter caseSensitive=false --> so lookups are caseInsensitive
PrefixMap<String> brandLookup = new StringPrefixMap<>(false);
brandLookup.put("GT-", "Samsung");
brandLookup.put("RM-", "Nokia");

String brandGT = brandLookup.getLongestMatch("GT-I8190N");   // --> "Samsung"
String brandRM = brandLookup.getLongestMatch("RM-1092");     // --> "Nokia"

All you need is to add this dependency to your project

<dependency>
  <groupId>nl.basjes.collections</groupId>
  <artifactId>prefixmap</artifactId>
  <version>2.0</version>
</dependency>

Version 2.0 If you are using Kryo 5.x you can use this to register all classes:

StringPrefixMap.configureKryo(kryo);

Blog post
=========
A bit more background about this datastructure and how it works can be found in this blog which I wrote about it:
Donations
=====
If this project has business value for you then don't hesitate to support me with a small donation.

[Donations via PayPal](https://www.paypal.me/nielsbasjes)

License
=======

Copyright (C) 2018-2020 Niels Basjes

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1174497368_1623906381.73/0/nielsbasjes-prefixmap-v2-0-0-g500ebea-tar-gz/nielsbasjes-
  prefixmap-e34e4cb/README.md
No license file was found, but licenses were detected in source scan.

2020 Niels Basjes
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~
~ https://www.apache.org/licenses/LICENSE-2.0

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.

Found in path(s):
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.814 jnr-constants 0.9.9

1.815 annotations 13.0
1.815.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2006 Sascha Weinreuter
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
 * /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-
jar/org/intellij/lang/annotations/Subst.java
 * /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-
jar/org/intellij/lang/annotations/Pattern.java
 * /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-
jar/org/intellij/lang/annotations/Language.java
 * /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-
jar/org/intellij/lang/annotations/RegExp.java
 * /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-
jar/org/intellij/lang/annotations/Identifier.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2012 JetBrains s.r.o.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/TestOnly.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/intellij/lang/annotations/MagicConstant.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/intellij/lang/annotations/JdkConstants.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/NotNull.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/Nullable.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/Nls.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2009 JetBrains s.r.o.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/NonNls.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/Nullable.java
* /opt/cola/permits/1138123144_1614251056.08/0/annotations-13-0-sources-3-jar/org/jetbrains/annotations/Nls.java
* No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2013 JetBrains s.r.o.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

1.816 parquet-jackson 1.12.0
1.816.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

--------------------------------------------------------------------------------

This product depends on Jackson and includes it in this binary artifact. Jackson is a high-performance JSON processor.

Copyright: 2007-2015 Tatu Saloranta and other contributors
Home page: http://jackson.codehaus.org/
License: http://www.apache.org/licenses/LICENSE-2.0.txt

1.817 json5 0.5.1

1.817.1 Available under license :

MIT License

Copyright (c) 2012-2016 Aseem Kishore, and [others](https://github.com/aseemk/json5/contributors).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.818 avatica-metrics 1.17.0
1.818.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
teditorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Calcite -- Avatica
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
1.819 lodash.escaperegexp 4.1.2 4.1.2

1.819.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.820 formidable 1.2.2
1.820.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011-present Felix Geisendrfer, and contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.821 graphlib 2.1.8 2.1.8
1.821.1 Available under license:
Copyright (c) 2012-2014 Chris Pettitt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.822 type-is 1.6.18 1.6.18
1.822.1 Available under license :
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.823 jacoco-core 0.7.9
1.823.1 Available under license :
NOT PROVIDED

1.824 is-data-descriptor 1.0.0 1.0.0
1.824.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.825 delayed-stream 1.0.0 1.0.0

1.825.1 Available under license:
Copyright (c) 2011 Debuggable Limited <felix@debuggable.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.826 libxdmcp 1.1.2-1.1
1.826.1 Available under license:
Copyright 1989, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

Author: Keith Packard, MIT X Consortium

1.827 caniuse-lite 1.0.30000874 1.0.30000874
1.827.1 Available under license:
Attribution 4.0 International

Creative Commons Corporation ("Creative Commons") is not a law firm and does not provide legal services or legal advice. Distribution of Creative Commons public licenses does not create a lawyer-client or other relationship. Creative Commons makes its licenses and related information available on an "as-is" basis. Creative Commons gives no warranties regarding its licenses, any material licensed under their terms and conditions, or any related information. Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible.

Using Creative Commons Public Licenses

Creative Commons public licenses provide a standard set of terms and conditions that creators and other rights holders may use to share original works of authorship and other material subject to copyright
and certain other rights specified in the public license below. The following considerations are for informational purposes only, are not exhaustive, and do not form part of our licenses.

Considerations for licensors: Our public licenses are intended for use by those authorized to give the public permission to use material in ways otherwise restricted by copyright and certain other rights. Our licenses are irrevocable. Licensors should read and understand the terms and conditions of the license they choose before applying it. Licensors should also secure all rights necessary before applying our licenses so that the public can reuse the material as expected. Licensors should clearly mark any material not subject to the license. This includes other CC-licensed material, or material used under an exception or limitation to copyright. More considerations for licensors: wiki.creativecommons.org/Considerations_for_licensors

Considerations for the public: By using one of our public licenses, a licensor grants the public permission to use the licensed material under specified terms and conditions. If the licensor's permission is not necessary for any reason—for example, because of any applicable exception or limitation to copyright—then that use is not regulated by the license. Our licenses grant only permissions under copyright and certain other rights that a licensor has authority to grant. Use of the licensed material may still be restricted for other reasons, including because others have copyright or other rights in the material. A licensor may make special requests, such as asking that all changes be marked or described. Although not required by our licenses, you are encouraged to respect those requests where reasonable. More considerations for the public: wiki.creativecommons.org/Considerations_for_licensees

=======================================================================

Creative Commons Attribution 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.
Section 1 -- Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.

b. Adapter’s License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

f. Licensed Material means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

g. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. Licensor means the individual(s) or entity(ies) granting rights under this Public License.

i. Share means to provide material to the public by any means or
process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

k. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

Section 2 -- Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

   a. reproduce and Share the Licensed Material, in whole or in part; and

   b. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License,
simply making modifications authorized by this Section 2(a) (4) never produces Adapted Material.

5. Downstream recipients.

a. Offer from the Licensor -- Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

b. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.

3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 -- License Conditions.
Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

1. If You Share the Licensed Material (including in modified form), You must:

   a. retain the following if it is supplied by the Licensor with the Licensed Material:

      i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);

      ii. a copyright notice;

   iii. a notice that refers to this Public License;

   iv. a notice that refers to the disclaimer of warranties;

   v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;

   b. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and

   c. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.

2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or hyperlink to a resource that includes the required information.

3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.

4. If You Share Adapted Material You produce, the Adapter's License You apply must not prevent recipients of the Adapted Material from complying with this Public License.
Section 4 -- Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;

b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and

c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 -- Disclaimer of Warranties and Limitation of Liability.

a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE. WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.

b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.

c. The disclaimer of warranties and limitation of liability provided
above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 -- Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:

1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

2. upon express reinstatement by the Licensor.

For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 -- Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.

b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 -- Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose
conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.

d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the Licensor. The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark “Creative Commons” or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

Creative Commons may be contacted at creativecommons.org.

1.828 pkg-dir 3.0.0
1.828.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.829 append-field 1.0.0 1.0.0

1.829.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Linus Unnebeck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.830 babel-plugin-dynamic-import-node 2.3.3
1.830.1 Available under license:
MIT License

Copyright (c) 2016 Airbnb

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.831 error_prone_annotations 2.3.4
1.831.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/DoNotMock.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
 * Copyright 2017 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/concurrent/GuardedBy.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/CheckReturnValue.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/DoNotCall.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Error Prone Authors.
*LICENSE
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/RequiredModifiers.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/Var.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/ForOverride.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/concurrent/LazyInit.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/CompileTimeConstant.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-jar/com/google/errorprone/annotations/Immutable.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
1.832 is-extglob 1.0.0

1.832.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.833 scheduler 0.19.1 0.19.1

1.833.1 Available under license:
MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.834 at-spi2-core 2.18.3-4ubuntu1

1.834.1 Available under license:

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain
that everyone understands that there is no warranty for this free
library. If the library is modified by someone else and passed on, we
want its recipients to know that what they have is not the original
version, so that any problems introduced by others will not reflect on
the original authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that companies distributing free
software will individually obtain patent licenses, thus in effect
transforming the program into proprietary software. To prevent this,
we have made it clear that any patent must be licensed for everyone's
free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary
GNU General Public License, which was designed for utility programs. This
license, the GNU Library General Public License, applies to certain
designated libraries. This license is quite different from the ordinary
one; be sure to read it in full, and don't assume that anything in it is
the same as in the ordinary license.

The reason we have a separate public license for some libraries is that
they blur the distinction we usually make between modifying or adding to a
program and simply using it. Linking a program with a library, without
changing the library, is in some sense simply using the library, and is
analogous to running a utility program or application program. However, in
a textual and legal sense, the linked executable is a combined work, a
derivative of the original library, and the ordinary General Public License
treats it as such.

Because of this blurred distinction, using the ordinary General
Public License for libraries did not effectively promote software
sharing, because most developers did not use the libraries. We
concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the
users of those programs of all benefit from the free status of the
libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based
on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact...
that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307 USA.
Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.835 aws-java-sdk-sts 1.11.375

1.835.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
jar/com/amazonaws/auth/SessionCredentialsHolder.java
*/opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
jar/com/amazonaws/auth/RefreshableTask.java
*/opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
jar/com/amazonaws/auth/ShouldDoAsyncSessionRefresh.java
*/opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
jar/com/amazonaws/auth/DaemonThreadFactory.java
*/opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
jar/com/amazonaws/auth/ShouldDoBlockingSessionRefresh.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
  jar/com/amazonaws/services/securitytoken/internal/STSProfileCredentialsService.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2013-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-
  jar/com/amazonaws/auth/policy/STSSecurityPolicy.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2012-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
Open Source Used In DNAC Platform Arica 2015

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-jar/com/amazonaws/auth/SessionCredentialsProviderFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-jar/com/amazonaws/auth/STSSessionCredentials.java
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-jar/com/amazonaws/auth/STSSessionCredentialsProvider.java
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-jar/com/amazonaws/auth/STSAssumeRoleSessionCredentialsProvider.java
* /opt/cola/permits/1180370927_1626285908.9/0/aws-java-sdk-sts-1-11-375-sources-1-jar/com/amazonaws/auth/WebIdentityFederationSessionCredentialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at

http://aws.amazon.com/apache2.0
1.836.1 Available under license:

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
* by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
   where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall
apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness
for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients’ rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),
version(s), and exceptions or additional permissions here)."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

### The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis...
or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other
recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

      (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot distribute
so as to satisfy simultaneously your obligations under this License and
any other pertinent obligations, then as a consequence you may not
distribute the Program at all. For example, if a patent license would
not permit royalty-free redistribution of the Program by all those who
receive copies directly or indirectly through you, then the only way you
could satisfy both it and this License would be to refrain entirely from
distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is implemented
by public license practices. Many people have made generous
contributions to the wide range of software distributed through that
system in reliance on consistent application of that system; it is up to
the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be
a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

1.837 jsreport-public-templates 2.2.0 2.2.0

1.837.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the
function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of

---

Open Source Used In DNAC Platform Arica 2036
the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version
of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.838 metrics-servlets 4.1.0

1.839 debug 3.2.7 3.2.7

1.839.1 Available under license :
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.840 exit-hook 1.1.1

1.840.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.841 batik-bridge 1.13

1.841.1 Available under license :

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Licensed to the Apache Software Foundation (ASF) under one or more
ccontributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/Messages.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/UserAgentViewport.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/SVGFfeOffsetElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/SVGBrokenLinkProvider.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/ConcreteTextPainter.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/og/apache/batik/bridge/UserAgent.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGCircleElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/GVTBuilder.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/GlyphLayout.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGDocumentBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/UpdateManager.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/CSSFontFace.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/GraphicsNodeBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGEllipseElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/DocumentBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGLineElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGSetElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/ScriptingEnvironment.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/EventTargetWrapper.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/FocusManager.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/Extension.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGFeElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/BaseScriptingEnvironment.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGSetElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/ScriptingEnvironment.java
Open Source Used In DNAC Platform Arica  2944

* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGFeImageElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/UserAgentAdapter.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/TextHit.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/TextUtilities.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/ClipBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/SVGFlowRootElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/BindableElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGUseElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGAnimateColorElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGAnimationElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGPatternElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/AbstractContentSelector.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/BindingListener.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/FlowGlyphLayout.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGTextPathElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/BridgeException.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/AnimatableSVGBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/FontFamilyResolver.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/ContentManager.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVG12RhinoInterpreter.java
Open Source Used In DNAC Platform Arica 2945

* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGSVGElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/GlobalWrapper.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/ConcreteTextSelector.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/svg12/SVG12ScriptingEnvironment.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/FlowTextNode.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/TextPainter.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGBridgeExtension.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/MultiResGraphicsNode.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/svg12/DefaultXBLManager.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGLinearGradientElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/Base64Consumer.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/UnitProcessor.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/svg12/SVG12BridgeEventSupport.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGDecoratedShapeElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/Window.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/PaintBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGA1tGlyphHandler.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/GenericBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/ViewBox.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/EmbeddedScriptSecurity.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/SVGUIutilities.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1.13-sources-1-jar/org/apache/batik/bridge/UpdateManagerAdapter.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/SVGTextElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/CursorManager.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/ConcreteTextLayoutFactory.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/FilterPrimitiveBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/InterruptedBridgeException.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/DefaultContentSelector.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/SVGAltGlyphElementBridge.java
* /opt/cola/permits/1146878544_1616835051.89/0/batik-bridge-1-13-sources-1-jar/org/apache/batik/bridge/svg12/SVG12URIResolver.java

1.842 jackson 1.9.13
1.842.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and
If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
1.843 string_decoder 0.10.31 0.10.31

1.843.1 Available under license:

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the
following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
USE OR OTHER DEALINGS IN THE SOFTWARE.

1.844 slf4j 1.7.28

1.844.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 * *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 * *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*
*/

Found in path(s):
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/spi/LoggerFactoryBinder.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/helpers/NOPMDCAdapter.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/ILoggerFactory.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/spi/MarkerFactoryBinder.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/Marker.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/MDC.java
* /opt/cola/permits/211276241_1646171445.23/0/slf4j-api-1-7-28-sources-jar/org/slf4j/helpers/NOPLogger.java
1.845 readline2 1.0.1

1.845.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.846 twill-discovery-core 0.6.0-incubating

1.846.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 */
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/InMemoryDiscoveryService.java
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/package-info.java
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/DefaultServiceDiscovered.java
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/ZKDiscoveryService.java
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/DiscoverableWrapper.java
* /opt/cola/permits/1137747390_1614121768.73/0/twill-discovery-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/DiscoverableAdapter.java

1.847 bootstrap 3.4.1 3.4.1

1.847.1 Available under license:
Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN
ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES
REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR
DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE
COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY
COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS
AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE
TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY
BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS
CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND
CONDITIONS.
1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are
assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The
original work was translated from English to Spanish,” or a modification could indicate “The original work has been modified.”;
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this
License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author,” or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert,
as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same
terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior
written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.
The MIT License (MIT)

Copyright (c) 2011-2019 Twitter, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.848 object-inspect 1.8.0

1.848.1 Available under license :

MIT License

Copyright (c) 2013 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.849 aws-java-sdk-core 1.11.375

1.849.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 * <p/>
 * http://aws.amazon.com/apache2.0
 * <p/>
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/partitions/model/Partitions.java

No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/log/InternalLogApi.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/EnvironmentVariableCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/http/IdleConnectionReaper.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/DefaultAWSCredentialsProviderChain.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/InstanceProfileCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/AWSCredentialsProviderChain.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/CRC32MismatchException.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/CustomBackoffStrategy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/DynamoDBBackoffStrategy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/EC2MetadataClient.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/ClasspathPropertiesFileCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/HandlerContextAware.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/AwsHostNameUtils.java
No license file was found, but licenses were detected in source scan.

/ *
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
* *
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* *
* /

Found in path(s):
/ *
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/auth/SystemPropertiesCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/util/BasicNameValuePair.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/util/URLEncodedUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/util/NameValuePair.java
* http://www.apache.org/licenses/LICENSE-2.0
*/

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* *
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* *
* http://aws.amazon.com/apache2.0

Open Source Used in DNAC Platform Arica 2964
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/transform/LegacyErrorUnmarshall.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/transform/Unmarshaller.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/auth/SigningAlgorithm.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/AmazonClientException.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/transform/SimpleTypeUnmarshallers.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/util/XmlUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/internal/http/CompositeErrorCodeParser.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License"). You may not
* use this file except in compliance with the License. A copy of the License is
* located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/transform/Unmarshaller.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/http/RepeatableInputStreamRequestEntity.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/config/InternalConfig.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/event/ProgressEventType.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/http/conn/SdkPlainSocketFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/policy/conditions/ConditionFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/VersionInfoUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/AmazonHttpClient.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/AmazonWebServiceClient.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/AWS3Signer.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/AWSServiceMetrics.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/ResponseMetadata.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/SignerTypeAware.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/handlers/RequestHandler2Adaptor.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/event/ProgressListener.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/ApacheHttpClientConfig.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/metrics/ServiceMetricCollector.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/metrics/ServiceMetricType.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/RetryUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/waiters/AcceptorPathMatcher.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/policy/conditions/ArnCondition.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/BasicAWSCredentials.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/internal/AuthErrorRetryStrategy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/handlers/HandlerAfterAttemptContext.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/AnonymousAWSCredentials.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/waiters/PollingStrategyContext.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/waiters/CompositeAcceptor.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/policy/Condition.java
Open Source Used In DNAC Platform Arica 2970

* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/AmazonWebServiceRequest.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/auth/policy/Action.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/NamespaceRemovingInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/AWSRequestMetricsFullSupport.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/waiters/MaxAttemptsRetryStrategy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/StringUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/SimpleMetricType.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/MetricInputStreamEntity.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/MetricType.java

---

**Open Source Used In DNAC Platform Arica 2970**

* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/AmazonWebServiceRequest.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/auth/policy/Action.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/NamespaceRemovingInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/AWSRequestMetricsFullSupport.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/waiters/MaxAttemptsRetryStrategy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/util/StringUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/SimpleMetricType.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/MetricInputStreamEntity.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1.11.375-sources-1.jar/com/amazonaws/metrics/MetricType.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/profile/path/AwsDirectoryBasePathProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/profile/path/AwsProfileFileLocationProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/json/internal/NullAsEmptyBodyProtocolRequestMarshaller.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/client/AwsAsyncClientParams.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/json/JsonContentTypeResolverImpl.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/retry/v2/RetryPolicy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/json/JsonContent.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/regions/DefaultAwsRegionProviderChain.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/http/timers/client/SdkInterruptedException.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/StructuredPojo.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/OperationInfo.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/json/internal/SimpleTypeJsonMarshallers.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/protocol/json/internal/JsonMarshallerContext.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/retry/v2/SimpleRetryPolicy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/retry/v2/BackoffStrategy.java

Open Source Used In DNAC Platform Arica  2973
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/regions/InstanceMetadataRegionProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/regions/AwsEnvVarOverrideRegionProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/builder/ExecutorFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/DefaultValueSupplier.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/internal/SimpleTypePathMarshallers.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/transform/MapUnmarshaller.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/internal/ValueToStringConverters.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/builder/AwsSyncClientBuilder.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/v2/MaxNumberOfRetriesCondition.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/RequestConfig.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/SdkHttpMetadata.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/regions/AwsRegionProviderChain.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/v2/AndRetryCondition.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/profile/path/AwsProfileFileLocationProviderChain.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/AWSStaticCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/ClientExecutionParams.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/SdkHttpMetadata.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/profile/internal/ProfileStaticCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/StaticSignerProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/v2/FixedDelayBackoffStrategy.java

Open Source Used In DNAC Platform Arica 2974
Open Source Used In DNAC Platform Arica 2976

* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/profile/internal/AwsProfileNameLoader.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/profile/path/config/SharedConfigDefaultLocationProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/SdkClock.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/ClientHandler.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/presign/PresignerFacade.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/request/HttpRequestFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/DefaultMarshallingType.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/ClientConfigurationFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/auth/profile/internal/ProfileAssumeRoleCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/internal/AmazonWebServiceRequestAdapter.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/ProtocolRequestMarshaller.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/util/TimingInfoUnmodifiable.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/AmazonWebServiceResult.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/builder/AwsAsyncClientBuilder.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/client/ClientHandlerParams.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/JsonClientMetadata.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/profile/path/cred/CredentialsDefaultLocationProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/regions/AwsSystemPropertyRegionProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/AwsErrorResponseHandler.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/retry/v2/RetryPolicyContext.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/internal.SdkFunction.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/util/TimingInfoFullSupport.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/JodaTime.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * 
 *   http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/StringMapBuilder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * 
 *   http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/SdkSSLContext.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/event/RequestProgressInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/profile/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/adapter/types/StringToByteBufferAdapter.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/adapter/types/TypeAdapter.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/adapter/types/StringToInputStreamAdapter.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/internal/auth/DefaultSignerProvider.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/internal/auth/SignerProviderContext.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/adapter/types/TypeAdapter.java
* /opt/cola/permits/1135892258_1613626888.74/jar/com/amazonaws/adapter/types/StringToInputStreamAdapter.java
jar/com/amazonaws/internal/auth/NoOpSignerProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/util/NullResponseMetadataCache.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/internal/auth/SignerProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2018 Amazon Technologies, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
*
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/http/conn/ssl/TLSProtocol.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/http/conn/ssl/SdkTLSSocketFactory.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/PolicyUtils.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2012-2018 Amazon Technologies, Inc.
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt for applicable license terms and NOTICE.txt for applicable notices.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1
Open Source Used In DNAC Platform Arica  2983

jar/com/amazonaws/util/Md5Utils.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/regions/InMemoryRegionsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/partitions/PartitionMetadataProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/regions/LegacyRegionXmlLoadUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/partitions/model/CredentialScope.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/partitions/model/Region.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/regions/RegionMetadataFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/http/apache/client/impl/CRC32ChecksumResponseInterceptor.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/protocol/json/SdkIonGenerator.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/transform/SimpleTypeCborUnmarshallers.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/regions/InMemoryRegionImpl.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 */

/*
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 */

http://aws.amazon.com/apache2.0

/*
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

/*
 * Interface for providing AWS region information. Implementations are free to use any strategy for
 * providing region information.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/regions/AwsRegionProvider.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 */

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 */

http://aws.amazon.com/apache2.0
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES *
* OR CONDITIONS OF ANY KIND, either express or implied. See the *
* License for the specific language governing permissions and *
* limitations under the License. *
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/jmx/spi/JmxInfoProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/jmx/spi/SdkMBeanRegistry.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/metrics/MetricAdmin.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/auth/AWSSessionCredentials.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/auth/AWSSessionCredentialsProvider.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/util/ServiceClientHolderInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/util/json/Jackson.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/util/ClassLoaderHelper.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/RequestClientOptions.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/metrics/MetricAdminMBean.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/jmx/JmxInfoProviderSupport.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/auth/BasicSessionCredentials.java

No license file was found, but licenses were detected in source scan.

/*
 *
* Copyright (c) 2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
*  http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
* Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
* the License. A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
* and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/GuardedBy.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/Immutable.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/NotThreadSafe.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/SdkProtectedApi.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/SdkInternalApi.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/ThreadSafe.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/SdkTestInternalApi.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/annotation/Beta.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not
 * use this file except in compliance with the License. A copy of the License is
 * located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/conn/ssl/ShouldClearSslSessionPredicate.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/log/InternalLogFactory.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/client/impl/SdkHttpClient.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/util/IdempotentUtils.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/transform/SimpleTypeIonUnmarshallers.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/IonFactory.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/IonParser.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/util/FakeIOException.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/internal/DateTimeJsonSerializer.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/apache/client/impl/ApacheHttpClientFactory.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/timers/package-info.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/apache/client/impl/ApacheHttpRequestFactory.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/http/apache/utils/ApacheUtils.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/handlers/IRequestHandler2.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/transform/SimpleTypeIonUnmarshallers.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/IonFactory.java
  * /opt/cola/permissions/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1.jar/com/amazonaws/protocol/json/IonParser.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is
 * distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/protocol/json/SdkStructuredJsonFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/http/apache/client/impl/ApacheConnectionManagerFactory.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/util/CapacityManager.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/http/apache/client/impl/ConnectionManagerAwareHttpClient.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/protocol/json/internal/JsonProtocolMarshaller.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
  jar/com/amazonaws/regions/Regions.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the 'License').
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the 'license' file accompanying this file. This file is distributed
 * on an 'AS IS' BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/util/CredentialUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt for applicable license terms and NOTICE.txt for applicable notices.
 * Licensed under the Apache License, Version 2.0 (the 'License').
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the 'license' file accompanying this file. This file is distributed
 * on an 'AS IS' BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-
jar/com/amazonaws/internal/ReleasableInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/ResettableInputStream.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */
/**
 * Interface for providing AWS credentials. Implementations are free to use any
 * strategy for providing AWS credentials, such as simply providing static
 * credentials that don't change, or more complicated implementations, such as
 * integrating with existing key management systems.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/AWSCredentialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/log/InternalLog.java
*/

* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
*
*    http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/MetricAware.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/jmx/SdkMBeanRegistrySupport.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/jmx/MBeans.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/Base32.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/MetricsInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/auth/RegionAwareSigner.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/HttpClientWrappingInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/AbstractBase32Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/Base32Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/EC2MetadataUtils.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/Base64Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/AbortedException.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/internal/SdkFilterInputStream.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/AbstractBase32Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/Base64Codec.java
* /opt/cola/permits/1135892258_1613626888.74/0/aws-java-sdk-core-1-11-375-sources-1-jar/com/amazonaws/util/SdkRuntime.java
1.850 calcite-linq4j 1.26.0
1.850.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Calcite
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.

The web site includes files generated by Jekyll.

1.851 babel-plugin-transform-es2015-unicode-regex 6.24.1
1.851.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.852 json-path 2.4.0

1.852.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/InvalidJsonException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/InvalidCriteriaException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/JsonPathException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/internal/path/PredicateContextImpl.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/Utils.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/internal/function/latebinding/PathLateBindingValue.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/json/JettisonProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/internal/path/CompiledPath.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/WriteContext.java
jar/com/jayway/jsonpath/spi/mapper/JacksonMappingProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/DocumentContext.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/json/JsonSmartJsonProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/JsonPath.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/mapper/MappingProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/mapper/GsonMappingProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/JsonFormatter.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/json/GsonJsonProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/MapFunction.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/PathNotFoundException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/evaluation/EvaluationContextImpl.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/evaluation/function/latebinding/ILateBindingValue.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/evaluation/path/PropertyPathToken.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/spi/json/GsonJsonProvider.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/PathNotFoundException.java
* /opt/cola/permits/1199743787_1630577685.84/0/json-path-2-4-0-sources-7-jar/com/jayway/jsonpath/MapFunction.java

1.853 netty-buffer 4.1.42.Final

1.853.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Open Source Used In DNAC Platform Arica 3008

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/Unpooled.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufInputstream.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolSubpage.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolThreadCache.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolChunk.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolChunkList.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractByteBufAllocator.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufUtil.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledByteBufAllocator.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufAllocator.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/CompositeByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledHeapByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ReadOnlyByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufOutputStream.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufOutputStream.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledByteBufAllocator.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/package-info.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/DuplicatedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ReadOnlyByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufOutputStream.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledByteBufAllocator.java
jar/io/netty/buffer/PoolArena.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-
jar/io/netty/buffer/SlicedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-
jar/io/netty/buffer/ByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-
jar/io/netty/buffer/ByteBufAllocatorMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-
jar/io/netty/buffer/PooledByteBufAllocatorMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-
jar/io/netty/buffer/ByteBufAllocatorMetricProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Open Source Used in DNAC Platform Arica 3010

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledDuplicatedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnsafeByteBufUtil.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolChunkMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolSubpageMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolArenaMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PoolChunkListMetric.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledUnsafeHeapByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledSlicedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnpooledUnsafeHeapByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractUnsafeSwappedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledHeapByteBuf.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 */

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledHeapByteBuf.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 */
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AdvancedLeakAwareByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/WrappedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ReadOnlyByteBufferBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/EmptyByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/FixedCompositeByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ReadOnlyUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractReferenceCountedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/DefaultByteBufHolder.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractDerivedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufProcessor.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/ByteBufHolder.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledUnsafeDirectByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/UnreleasableByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/SimpleLeakAwareByteBuf.java

No license file was found, but licenses were detected in source scan.

/*. No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance

Jar/io/netty/buffer/AbstractPooledDerivedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/AbstractUnpooledSlicedByteBuf.java
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/io/netty/buffer/PooledDuplicatedByteBuf.java

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
#   http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/META-INF/native-image/io.netty/buffer/native-image.properties

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1144317897_1615901241.09/0/netty-buffer-4-1-42-final-sources-3-jar/META-INF/maven/io.netty/netty-buffer/pom.xml

1.854 acorn 3.3.0
1.854.1 Available under license :

Copyright (C) 2012-2016 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.855 hppc 0.7.1
1.855.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is unintentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013, Carrot Search s.c., Boznicza 11/56, Poznan, Poland

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

ACKNOWLEDGEMENT

=================

HPPC borrowed code, ideas or both from:

  (Apache license)
* Fastutil, http://fastutil.di.unimi.it/
  (Apache license)
* Koloboke, https://github.com/OpenHFT/Koloboke
  (Apache license)

1.856 log4j-api 2.17.1
1.856.1 Available under license :

Apache Log4j 1.x Compatibility API
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.857 js-tokens 4.0.0 4.0.0
1.857.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017, 2018 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.858 string_decoder 1.1.1 1.1.1
1.858.1 Available under license :
Node.js is licensed for use as follows:

---

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.859.1 Available under license:

You may use the Forge project under the terms of either the BSD License or the GNU General Public License (GPL) Version 2.

The BSD License is recommended for most projects. It is simple and easy to understand and it places almost no restrictions on what you can do with the Forge project.

If the GPL suits your project better you are also free to use Forge under that license.

You don’t have to do anything special to choose one license or the other and you don’t have to notify anyone which license you are using. You are free to use this project in commercial projects as long as the copyright header is left intact.

If you are a commercial entity and use this set of libraries in your commercial software then reasonable payment to Digital Bazaar, if you can afford it, is not required but is expected and would be appreciated. If this library saves you time, then it's saving you money. The cost of developing the Forge software was on the order of several hundred hours and tens of thousands of dollars. We are attempting to strike a balance between helping the development community while not being taken advantage of by lucrative commercial entities for our efforts.

New BSD License (3-clause)
Copyright (c) 2010, Digital Bazaar, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Digital Bazaar, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DIGITAL BAZAAR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free 
software. If the software is modified by someone else and passed on, we 
want its recipients to know that what they have is not the original, so 
that any problems introduced by others will not reflect on the original 
authors' reputations.

Finally, any free program is threatened constantly by software 
patents. We wish to avoid the danger that redistributors of a free 
program will individually obtain patent licenses, in effect making the 
program proprietary. To prevent this, we have made it clear that any 
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and 
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains 
a notice placed by the copyright holder saying it may be distributed 
under the terms of this General Public License. The "Program", below, 
refers to any such program or work, and a "work based on the Program" 
means either the Program or any derivative work under copyright law: 
that is to say, a work containing the Program or a portion of it, 
either verbatim or with modifications and/or translated into another 
language. (Hereinafter, translation is included without limitation in 
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not 
covered by this License; they are outside its scope. The act of 
running the Program is not restricted, and the output from the Program 
is covered only if its contents constitute a work based on the 
Program (independent of having been made by running the Program). 
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's 
source code as you receive it, in any medium, provided that you 
conspicuously and appropriately publish on each copy an appropriate 
copyright notice and disclaimer of warranty; keep intact all the 
notices that refer to this License and to the absence of any warranty; 
and give any other recipients of the Program a copy of this License 
along with the Program.

You may charge a fee for the physical act of transferring a copy, and 
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion 
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of
the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.
1.860.1 Available under license:

Apache HttpCore
Copyright 2005-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.861 curvesapi 1.06
1.861.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005, Graph Builder
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * * Redistributions of source code must retain the above copyright notice,
 * this list of conditions and the following disclaimer.
 * *
 * * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 *
 * * Neither the name of Graph Builder nor the names of its contributors may be
 * used to endorse or promote products derived from this software without
 * specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 */
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
* CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/AcoshFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/ExpressionTree.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/SqrtFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/FloorFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/CombinFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/curve/CubicBSpline.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/MultNode.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/curve/LagrangeCurve.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/FuncNode.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/TanFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/TanhFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/VarNode.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/PowFunction.java
* /opt/cola/permits/1003993495_1646171700.49/0/curvesapi-1-06-sources-jar/com/graphbuilder/math/func/EFunction.java

Open Source Used In DNAC Platform Arica  3037
No license file was found, but licenses were detected in source scan.

* Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0 *

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and limitations under the License. */

No license file was found, but licenses were detected in source scan.

(The MIT License)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

No license file was found, but licenses were detected in source scan.

* /opt/cola/permits/1239170125_1641852247.74/0/growl-1-10-5-tgz/package/Readme.md
No license file was found, but licenses were detected in source scan.

// Growl - Copyright TJ Holowaychuk <tj@vision-media.ca> (MIT Licensed)

Found in path(s):
* /opt/cola/permits/1239170125_1641852247.74/0/growl-1-10-5-tgz/package/lib/growl.js

**1.863 classmate 0.8.0**

**1.863.1 Available under license :**
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses.
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.864 is-regex 1.1.1

1.864.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.865 end-of-stream 1.4.4

1.865.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.866 collection-visit 1.0.0

1.866.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.867 glob-base 0.3.0
1.867.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.868 react-dom 16.4.1 16.4.1
1.868.1 Available under license:
MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.869 babel-helper-explode-assignable-expression 6.24.1

1.869.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTicular PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
 DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
“Software”), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.870 netty-codec-socks 4.1.42.Final

1.870.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 *
*/

Found in path(s):

No license file was found, but licenses were detected in source scan.
~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2013 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-jar/io/netty/handler/codec/socks/SocksMessageType.java
jar/io/netty/handler/codec/socks/SocksAddressType.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/SocksVersion.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/v5/Socks5PasswordAuthStatus.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksCmdType.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksCmdStatus.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksRequestType.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksResponseType.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/v5/Socks5CommandStatus.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/v5/Socks5AddressType.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/UnknownSocksRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/v5/DefaultSocks5CommandRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socksx/v5/DefaultSocks5CommandRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/SocksRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
jar/io/netty/handler/codec/socks/UnknownSocksRequest.java
* /opt/cola/permits/1258272163_1643050048.65/0/netty-codec-socks-4-1-42-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
1.871 string Prototype TrimEnd 1.0.1

1.871.1 Available under license :

MIT License

Copyright (c) 2017 Khaled Al-Ansari

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.872 string-width 3.1.0

1.872.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.873 slf4j-ext 1.7.25

1.874 jersey 2.5.1

1.874.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.
1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License.
You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software.
against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration
regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

---------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.


The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all
the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show

them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you

legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no

warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to

know that what they have is not the original, so that any problems introduced by others will not reflect on the

original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that

redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To

prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying

it may be distributed under the terms of this General Public License. The "Program", below, refers to any such

program or work, and a "work based on the Program" means either the Program or any derivative work under

copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications

and/or translated into another language. (Hereinafter, translation is included without limitation in the term

"modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its

scope. The act of running the Program is not restricted, and the output from the Program is covered only if its

contents constitute a work based on the Program (independent of having been made by running the Program). Whether

that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium,

provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and

disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and

give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty

protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the

Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you

also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of
      any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Open Source Used In DNAC Platform Arica 3065

Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and
distribute the resulting executable under terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that module. An independent module is a module
which is not derived from or based on this library. If you modify this library, you may extend this exception to your
version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement
from your version.

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.
 *
 * The contents of this file are subject to the terms of either the GNU
 * General Public License Version 2 only ("GPL") or the Common Development
 * and Distribution License("CDDL") (collectively, the "License"). You
 * may not use this file except in compliance with the License. You can
 * obtain a copy of the License at
 * http://glassfish.java.net/public/CDDL+GPL_1_1.html
 * or packager/legal/LICENSE.txt. See the License for the specific
 * language governing permissions and limitations under the License.
 *
 * When distributing the software, include this License Header Notice in each
 * file and include the License file at packager/legal/LICENSE.txt.
 *
 * GPL Classpath Exception:
 * Oracle designates this particular file as subject to the "Classpath"
 * exception as provided by Oracle in the GPL Version 2 section of the License
 * file that accompanied this code.
 *
 * Modifications:
 * If applicable, add the following below the License Header, with the fields
 * enclosed by brackets [] replaced by your own identifying information:
 * "Portions Copyright [year] [name of copyright owner]"
 *
 * Contributor(s):
 * If you wish your version of this file to be governed by only the CDDL or
 * only the GPL Version 2, indicate your decision by adding "[Contributor]
 * elects to include this software in this distribution under the [CDDL or GPL
 * Version 2] license." If you don't indicate a single choice of license, a
 * recipient has the option to distribute your version of this file under
 * either the CDDL, the GPL Version 2 or to extend the choice of license to
 * its licensees as provided above. However, if you add GPL Version 2 code
 * and therefore, elected the GPL Version 2 license, then the option applies
 * only if the new code is made subject to such option by the copyright
 * holder.
 */

DO NOT TRANSLATE OR LOCALIZE.

%% The following software may be included in this product: ASM
Use of any of this software is governed by the terms of the license below:

Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

%% The following software may be included in this product: Jettison
Use of any of this software is governed by the terms of the license below:

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously
marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

%%% The following software may be included in this product: Jackson
Use of any of this software is governed by the terms of the license below:

Jackson is dual-licensed under two alternative popular Open Source licenses: Apache (AL 2.0) and Gnu Lesser GPL (LGPL 2.1). You choose one or the other, as necessary (if you want to redistribute the code for use, you do not need license), and abide by the license rules as defined by the respective license agreement (and only that one).

---------------------------------------------------------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright

Open Source Used In DNAC Platform Arica 3074
owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written
communication sent to the Licensor or its representatives, including but not limited to communication on electronic
mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously
marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been
received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section)
patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent
infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date
such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any
medium, with or without modifications, and in Source or Object form, provided that You meet the following
conditions:

   1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

   2. You must cause any modified files to carry prominent notices stating that You changed the files; and

   3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent,
      trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to
      any part of the Derivative Works; and

   4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You
distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding
those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a
NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided
along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such
third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not
modify the License. You may add Your own attribution notices within Derivative Works that You distribute,
alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices
cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
No license file was found, but licenses were detected in source scan.

// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,
// agreement (which includes confidentiality provisions). These features may

Found in path(s):
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-
jar/google/api/launch_stage.proto

No license file was found, but licenses were detected in source scan.

/**
 * <pre>
 * Early Access features are limited to a closed group of testers. To use
 * these features, you must sign up in advance and sign a Trusted Tester
 * agreement (which includes confidentiality provisions). These features may
 * be unstable, changed in backward-incompatible ways, and are not
 * guaranteed to be released.
 * </pre>
 *
 * <code>EARLY_ACCESS = 1;</code>
 */

Found in path(s):
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-
jar/com/google/api/LaunchStage.java

No license file was found, but licenses were detected in source scan.

// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License.
// You may obtain a copy of the License at
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-
jar/google/api/monitoring.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-
jar/google/api/control.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-
jar/google/cloud/audit/audit_log.proto

* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/geo/type/viewport.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/rpc/error_details.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/field_behavior.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/consumer.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/logging/type/log_severity.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/system_parameter.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/label.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/monitored_resource.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/endpoint.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/usage.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/type/color.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/client.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/longrunning/operations.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/billing.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/metric.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/config_change.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/type/timeofday.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/annotations.proto
* /opt/cola/permits/1199767240_1630583079.56/0/proto-google-common-protos-1-17-0-sources-1-jar/google/api/service.proto

1.876 batik-script 1.13
1.876.1 Available under license:

No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

# This file registers the interpreter factories for the
# script languages that Batik understands.
#
# $Id: org.apache.batik.script.InterpreterFactory 1733416 2016-03-03 07:07:13Z gadams $
# ******************************************************************************

# ECMA Script
org.apache.batik.bridge.RhinoInterpreterFactory

# TCL
# org.apache.batik.script.jacl.JaclInterpreterFactory

# Python
# org.apache.batik.script.jpython.JPythonInterpreterFactory

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878633_1616810515.71/0/batik-script-1-13-sources-1-jar/org/apache/batik/script/InterpreterFactory.java
* /opt/cola/permits/1146878633_1616810515.71/0/batik-script-1-13-sources-1-jar/org/apache/batik/script/InterpreterException.java
* /opt/cola/permits/1146878633_1616810515.71/0/batik-script-1-13-sources-1-jar/org/apache/batik/script/ImportInfo.java
* /opt/cola/permits/1146878633_1616810515.71/0/batik-script-1-13-sources-1-jar/org/apache/batik/script/jacl/JaclInterpreter.java
1.877 curator-recipes 5.1.0

1.877.1 Available under license:

Curator Recipes
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.878 kerb-core 1.0.0
1.878.1 Available under license :

Kerby-kerb core
Copyright 2014-2017 The Apache Software Foundation
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.879 groovy-sql 3.0.8
1.879.1 Available under license :
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing,
software distributed under the License is distributed on an
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License for the
specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-jar/groovy/sql/package.html
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/ResultSetOutParameter.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/ExtractIndexAndSql.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/ResultSetMetaDataWrapper.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/GroovyResultSetExtension.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/Sql.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/BatchingPreparedStatementWrapper.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/CallResultSet.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/DataSet.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/GroovyRowResult.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/SqlOrderByVisitor.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/SqlWithParams.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/InOutParameter.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/groovy/sql/InParameter.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/org/apache/groovy/sql/extensions/SqlExtensions.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/org/apache/groovy/sql/extensions/SqlExtensions.java
* /opt/cola/permits/1218311877_1634925046.69/0/groovy-sql-3-0-8-sources-
jar/org/apache/groovy/sql/extensions/SqlExtensions.java

1.880 request 2.88.2 2.88.2
1.880.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.881 ignore-by-default 1.0.1

1.881.1 Available under license:
ISC License (ISC)
Copyright (c) 2016, Mark Wubben

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.882 user-home 2.0.0

1.882.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.883 chainsaw 0.1.0 0.1.0

1.883.1 Available under license:
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT license:
http://www.opensource.org/licenses/mit-license.php

1.884 jsreport-licensing 2.2.3 2.2.3

1.884.1 Available under license:
No license file was found, but licenses were detected in source scan.

    license: 'enterprise',
    license: body.license,
    license: definition.options.license,

    Found in path(s):
    * /opt/cola/permits/1103714594_1615295527.4/0/jsreport-licensing-2-2-3-tgz/package/lib/licensing.js
1.885 url-parse-lax 3.0.0

1.885.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.886 cross-spawn 7.0.3

1.886.1 Available under license:
The MIT License (MIT)

Copyright (c) 2018 Made With MOXY Lda <hello@moxy.studio>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.

1.887 archiver 3.0.0 3.0.0
1.887.1 Available under license:
Copyright (c) 2012-2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.888 coffee-script 2.5.1 2.5.1
1.888.1 Available under license:
Copyright (c) 2009-2018 Jeremy Ashkenas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESSION OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.889 string-replace-async 2.0.0 2.0.0
1.889.1 Available under license :
The MIT License (MIT)

Copyright (c) Dmitrii Sobolev <disobolev@icloud.com> (github.com/dsblv)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.890 browser-stdout 1.3.1
1.890.1 Available under license :
Copyright 2018 kumavis

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby
granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.891 cliui 6.0.0 6.0.0

1.891.1 Available under license:
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.892 listener-collection 1.2.0 1.2.0

1.892.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jan Blaha

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.893 http-cache-semantics 4.1.0

1.893.1 Available under license:

Copyright 2016-2018 Kornel Lesiski

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.894 metrics-json 4.0.2

1.895 semver 5.6.0 5.6.0

1.895.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.896 domelementtype 1.1.3 1.1.3

1.896.1 Available under license:

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.897 class-utils 0.3.6 0.3.6

1.897.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, 2017-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.898 node-environment-flags 1.0.6
1.898.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2009 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.
Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

## -*-makefile-*-

## BSD-specific setup (FreeBSD, OpenBSD, NetBSD, *BSD)
## Copyright (c) 1999-2009, International Business Machines Corporation and others. All Rights Reserved.

## Commands to generate dependency files
GEN_DEPS.c=$(CC) -E -MM $(DEFS) $(CPPFLAGS)
GEN_DEPS.cc=$(CXX) -E -MM $(DEFS) $(CPPFLAGS)

## Flags for position independent code
SHAREDLIBCFLAGS = -fPIC
SHAREDLIBCXXFLAGS = -fPIC
SHAREDLIBCPPFLAGS = -DPIC

## Additional flags when building libraries and with threads
THREADSCPPFLAGS = -D_REENTRANT
LIBCPPFLAGS =

## Compiler switch to embed a runtime search path
LD_RPATH=
LD_RPATH_PRE=-Wl,-rpath,

## Compiler switch to embed a library name
LD_SONAME = -Wl,-soname -Wl,$(notdir $(MIDDLE_SO_TARGET))

## Shared library options
LD_SOOPTIONS= -Wl,-Bsymbolic

## Shared object suffix
SO = so

## Non-shared intermediate object suffix
STATIC_O = ao

## Compilation rules
%.$(STATIC_O): $(srcdir)/%.c
$(COMPILE.c) $(STATICCPPFLAGS) $(STATICCFLAGS) -o $@ $<
%.o: $(srcdir)/%.c
$(COMPILE.c) $(DYNAMICCPPFLAGS) $(DYNAMICCFLAGS) -o $@ $<

%.$(STATIC_O): $(srcdir)/%.cpp
$(COMPILE.cc) $(STATICCPPFLAGS) $(STATICCXXFLAGS) -o $@ $<
%.o: $(srcdir)/%.cpp
$(COMPILE.cc) $(DYNAMICCPPFLAGS) $(DYNAMICCXXFLAGS) -o $@ $<

Open Source Used In DNAC Platform Arica 3108
## Dependency rules
%.d: $(srcreg)/%.c
@echo "generating dependency information for $<"
@sHELL) -ec "$(GEN_DEPS.c) $< \\n| sed '"sA($*)\(.*\)o \1\n\1.o $@ : /g" > $@; \\
[ -s $@ ] || rm -f $@'
%.d: $(srcreg)/%.cpp
@echo "generating dependency information for $<"
@sHELL) -ec "$(GEN_DEPS.cc) $< \\n| sed '"sA($*)\(.*\)o \1\n\1.o $@ : /g" > $@; \\
[ -s $@ ] || rm -f $@'

## Versioned libraries rules
%.$(SO).$(SO_TARGET_VERSION_MAJOR): %.$(SO).$(SO_TARGET_VERSION)
$(RM) $@ && ln -s ${<F} $@
%.$(SO): %.$(SO).$(SO_TARGET_VERSION_MAJOR)
$(RM) $@ && ln -s ${*F}.$(SO).$(SO_TARGET_VERSION) $@

## End FreeBSD-specific setup

1.900 guava 22.0-android

1.900.1 Available under license :
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2015 The Guava Authors*
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except *
* in compliance with the License. You may obtain a copy of the License at *
* *
* http://www.apache.org/licenses/LICENSE-2.0 *
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License *
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either *
* express *
* or implied. See the License for the specific language governing permissions and limitations under *
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/hash/MacHashFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/AgegregateFutureState.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/Platform.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/hash/LittleEndianByteArray.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/ConsumingQueueIterator.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/WellBehavedMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/GwtTransient.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/package-info.java

No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (C) 2008 The Guava Authors 
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except 
* in compliance with the License. You may obtain a copy of the License at 
* 
* http://www.apache.org/licenses/LICENSE-2.0 
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License 
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express 
* or implied. See the License for the specific language governing permissions and limitations under 
* the License. 
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/CharMatcher.java 
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/primitives/Bytes.java 
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/internal/Finalizer.java 
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/Converter.java 
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SortedMultisets.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SortedMultiset.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2005 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2011 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/*
 * This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
 * and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 *
 * As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
 * java.util.HashMap class.
 */

Found in path(s):
*/

/*
 * Copyright (C) 2013 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/reflect/TypeVisitor.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/Verify.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/VerifyException.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/eventbus/SubscriberExceptionContext.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/Runnables.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/CharSequenceReader.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/hash/HashingInputStream.java

No license file was found, but licenses were detected in source scan.

/"
* Copyright (C) 2016 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/AbstractGraph.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
Open Source Used In DNAC Platform Arica 3121

* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingImmutableList.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/TransformedIterator.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/EvictingQueue.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/AbstractNavigableMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingDeque.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/TreeTraverser.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/FilteredEntrySetMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/TreeRangeMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/AbstractMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingBlockingDeque.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/DescendingMultiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingImmutableMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/FilteredSetMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/FilteredMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingNavigableMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/ForwardingImmutableSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/BinaryTreeTraverser.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/FilteredEntryMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/AllEqualOrdering.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/SortedMultisetBridge.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/ForwardingNavigableSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/ImmutableNetwork.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/ImmutableGraph.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/MutableGraph.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/Network.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/TopKSelector.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/MutableNetwork.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/Network.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/TopKSelector.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/MutableNetwork.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/TopKSelector.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/graph/MutableNetwork.java

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/Atomics.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/Monitor.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/net/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/Equivalence.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/Strings.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/annotations/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ListeningExecutorService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/annotations/Beta.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/Ascii.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collection/SortedLists.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/ContiguousSet.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/Hashing.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
Open Source Used In DNAC Platform Arica 3125

* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/Predicates.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/EnumMultiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/Supplier.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/LineReader.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/Eventbus/DeadEvent.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/LineBuffer.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/base/Throwables.java

Open Source Used In DNAC Platform Arica 3125
//
// Copyright (C) 2008 The Guava Authors

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors

Open Source Used In DNAC Platform Arica 3126
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/StandardTable.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableBiMap.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/StandardRowSortedTable.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Serialization.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Range.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/CollectPreconditions.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Collections2.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableEntry.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableMapEntrySet.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Platform.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/HashBasedTable.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableMultiset.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableSortedSet.java
*/opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/EnumBiMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/MapDifference.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ListMultimap.java
Open Source Used In DNAC Platform Arica  3129

* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Interner.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingMultiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/TreeMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/AbstractMapBasedMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/AbstractMapEntry.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Multimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingMultimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ClassToInstanceMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/LexicographicalOrdering.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/TreeMultiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/RegularImmutableSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/NullsFirstOrdering.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/Maps.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingSortedSet.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/io/ByteSink.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/io/ByteSink.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/math/PairedStatsAccumulator.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/Multimap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/Multiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/io/FileWriteMode.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/io/ByteSource.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/hash/AbstractByteHasher.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/CartesianList.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/cache/LongAddable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/reflect/TypeCapture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/cache/LongAddables.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/math/LinearTransformation.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/math/Stats.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/CharSink.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/ServiceManager.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/reflect/ClassPath.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/xml/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/reflect/Parameter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableClassToInstanceMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ComparisonChain.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableAsList.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ArrayTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableEnumSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ImmutableTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/RegularImmutableSortedSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/ForwardingTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/collection/SingletonImmutableTable.java

No license file was found, but licenses were detected in source scan.
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/eventbus/SubscriberRegistry.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
/* Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/*
/**
 * Outer class that exists solely to let us write [\@code Partially.GwtIncompatible] instead of plain
 * [\@code GwtIncompatible]. This is more accurate for [\@link Futures#catching], which is available
 * under GWT but with a slightly different signature.
 *
 * \<p>\We can't use [\@code PartiallyGwtIncompatible] because then the GWT compiler wouldn't recognize
 * it as a [\@code GwtIncompatible] annotation. And for [\@code Futures.catching], we need the GWT
 * compiler to autostrip the normal server method in order to expose the special, inherited GWT
 * version.
 */
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/Partially.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2006 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/escape/CharEscaper.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/CaseFormat.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/reflect/TypeToken.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/annotations/VisibleForTesting.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/PatternFilenameFilter.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/Futures.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/io/AppendableWriter.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-jar/com/google/common/hash/package-info.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/DenseImmutableTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/escape/Platform.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SparseImmutableTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/net/UrlEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SpliceEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SpliceEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/io/LineProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/html/HtmlEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/UnsignedBytes.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/Calls.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/cache/LocalCache.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/Cut.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/escape/Escapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/SpliceEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/AbstractService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/reflect/TypeResolver.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/Splitter.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/Platform.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/MapMaker.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/Service.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/cache/CacheBuilder.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/annotations/GwtIncompatible.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/net/InternetDomainName.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/io/ByteArrayDataInput.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/MapMakerInternalMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collect/RegularImmutableTable.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/SignedBytes.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/annotations/GwtCompatible.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/io/ByteProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/xml/XmlEscapers.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/net/HostSpecifier.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 */
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/BloomFilterStrategies.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/HashFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/Funnel.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/cache/RemovalListeners.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/Queues.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/math/DoubleUtils.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/MessageDigestHashFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/Uninterruptibles.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/reflect/Types.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/hash/Hashing.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
  jar/com/google/common/collect/BoundType.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/ParseRequest.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/base/FunctionalEquivalence.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/RegularContiguousSet.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/math/MathPreconditions.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/math/DoubleMath.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/collection/RegularImmutableMultiset.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/cache/AbstractCache.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/UnsignedLongs.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/primitives/UnsignedInteger.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 * ...
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * ...
 * http://www.apache.org/licenses/LICENSE-2.0
 * ...
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088313_1594481589.37/0/guava-22-0-android-sources-
jar/com/google/common/graph/BaseGraph.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

1.901 superagent 2.3.0
1.901.1 Available under license :
(The MIT License)

Copyright (c) 2014-2016 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.902 should-type-adaptors 1.1.0 1.1.0

1.902.1 Available under license:
MIT License

Copyright (c) 2016 Should.js assertion library

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.903 wrappy 1.0.2 1.0.2
1.903.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.904 capnproto 0.5.3-2ubuntu1.1

1.904.1 Available under license:

Copyright (c) 2013-2014 Sandstorm Development Group, Inc. and contributors
Licensed under the MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# This file contains a list of people who've made non-trivial contribution to the Google C++ Testing Framework project. People who commit code to the project are encouraged to add their names here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@gmail.com>
Ji Sigursson <joi@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@gmail.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@google.com>
Patrick Riley <pfr@google.com>
Peter Kaminski <piotrkgoogle.com>
1.905 apiguardian-api 1.1.0

1.905.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1145177138_1616570429.4/0/apiguardian-api-1-1-0-sources-1-jar/org/apiguardian/api/API.java

1.906 jansi-native 1.8

1.906.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2017 the original author(s).
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
Found in path(s):
* /opt/cola/permits/1257860186_1643019499.32/0/jansi-native-1-8-sources-jar/org/fusesource/jansi/internal/WindowsSupport.java
* /opt/cola/permits/1257860186_1643019499.32/0/jansi-native-1-8-sources-jar/org/fusesource/jansi/internal/CLibrary.java
* /opt/cola/permits/1257860186_1643019499.32/0/jansi-native-1-8-sources-jar/org/fusesource/jansi/internal/Kernel32.java

1.907 bufferutil 1.2.1 1.2.1

1.907.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.908 to-fast-properties 1.0.3

1.908.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Petka Antonov
2015 Sindre Sorhus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.909 jackson-annotations 2.11.4

1.909.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.910 execa 1.0.0 1.0.0
**1.910.1 Available under license:**

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.911 crc32-stream 3.0.1 3.0.1**

**1.911.1 Available under license:**

Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.912 jesc 1.3.0

1.912.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.913 guice 4.0

1.913.1 Available under license:

Google Guice - Core Library
Copyright 2006-2015 Google, Inc.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.914 common-utils 6.1.1

1.914.1 Available under license:

The following libraries are included in packaged versions of this project:

* Apache ZooKeeper
  * COPYRIGHT: Copyright 2009-2014 The Apache Software Foundation
  * LICENSE: licenses/LICENSE.apache2.txt
  * NOTICE: licenses/NOTICE.zookeeper.txt
  * HOMEPAGE: http://zookeeper.apache.org/

* jline
  * COPYRIGHT: Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
  * LICENSE: licenses/LICENSE.bsd.txt
  * HOMEPAGE: http://jline.sourceforge.net/

* SLF4J
  * COPYRIGHT: Copyright (c) 2004-2013 QOS.ch
  * LICENSE: licenses/LICENSE.mit.txt
  * HOMEPAGE: http://www.slf4j.org/

* ZkClient
  * LICENSE: licenses/LICENSE.apache2.txt
  * HOMEPAGE: https://github.com/sgroschupf/zkclient

Apache ZooKeeper
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.915 jsreport-studio 2.10.1 2.10.1

1.915.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Erik Rasmussen

Open Source Used In DNAC Platform Arica 3171
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.916 binary-extensions 1.13.1

1.916.1 Available under license :
MIT License

Copyright (c) 2019 Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com), Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.917 @babel/code-frame 7.5.5 7.5.5
1.917.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.918 ajv 5.5.2 5.5.2

1.918.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.919 whatwg-fetch 3.4.1 3.4.1
1.919.1 Available under license:

Copyright (c) 2014-2016 GitHub, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.920 regjsparser 0.1.5 0.1.5
1.920.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
1.921 asm 2.2.0-b21

1.921.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2007 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
  jar/org/objectweb/asm/util/Traceable.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
  jar/org/objectweb/asm/util/ASMifiable.java

No license file was found, but licenses were detected in source scan.

2011 Oracle and/or its affiliates. All rights reserved.
The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can obtain a copy of the License at https://glassfish.dev.java.net/public/CDDL+GPL_1_1.html or packager/legal/LICENSE.txt. See the License for the specific language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields enclosed by brackets [] replaced by your own identifying information: "Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or only the GPL Version 2, indicate your decision by adding "[Contributor] elects to include this software in this distribution under the [CDDL or GPL Version 2] license." If you don't indicate a single choice of license, a recipient has the option to distribute your version of this file under either the CDDL, the GPL Version 2 or to extend the choice of license to its licensees as provided above. However, if you add GPL Version 2 code and therefore, elected the GPL Version 2 license, then the option applies only if the new code is made subject to such option by the copyright holder.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/pom.xml
No license file was found, but licenses were detected in source scan.

* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
  jar/org/objectweb/asm/xml/package.html
No license file was found, but licenses were detected in source scan.

ASM XML Adapter
Copyright (c) 2004, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
Contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/xml/asm-xml.dtd

No license file was found, but licenses were detected in source scan.

/**
 * ASM XML Adapter
 * Copyright (c) 2004, Eugene Kuleshov
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/xml/SAXFieldAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/xml/ASMContentHandler.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/xml/SAXClassAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/xml/SAXAdapter.java
No license file was found, but licenses were detected in source scan.

/** *
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2005 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/commons/InstructionAdapter.java

No license file was found, but licenses were detected in source scan.

2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/tree/analysis/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/attrs/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/util/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/signature/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/tree/package.html
  */opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-
jar/org/objectweb/asm/commons/package.html
No license file was found, but licenses were detected in source scan.

/***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/AnnnotationVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/AnnotationVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/ClassReader.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/RemappingSignatureAdapter.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SourceValue.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/TraceFieldVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/signature/SignatureWriter.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/TableSwitchInsnNode.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/ASMifierFieldVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/CheckSignatureAdapter.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/FieldVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/AnnotationVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/ClassReader.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/RemappingSignatureAdapter.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SourceValue.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/TraceAbstractVisitor.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/Edge.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SourceValue.java
*/opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
*/
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/Handler.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/VarInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/VarInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/IincInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/commons/RemappingMethodAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/util/CheckAnnotationAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/commons/RemappingAnnotationAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/InsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/Item.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/MethodInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/AnnotationWriter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/MethodVisitor.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/tree/TypeInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/util/ASMifierMethodVisitor.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/commons/EmptyVisitor.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1.jar/org/objectweb/asm/util/ASMifierAbstractVisitor.java

Open Source Used In DNAC Platform Arica 3182
Open Source Used In DNAC Platform Arica  3183

* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/ClassVisitor.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/TraceAnnotationVisitor.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/LocalVariablesSorter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/Analyzer.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/util/CheckFieldAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/LdcInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/TryCatchBlockNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/FieldInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/lookupswitchInsnNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/RemappingClassAdapter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/Frame.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/FieldWriter.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/tree/TryCatchBlockNode.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/Remapper.java
* /opt/ws_local/PERMITS_SQL/1048949653_1591433284.08/0/asm-all-repackaged-2-2-0-b21-sources-1-jar/org/objectweb/asm/commons/Method.java

Open Source Used In DNAC Platform Arica  3183
1.922 timed-out 2.0.0 2.0.0

1.922.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.923 babel-helper-optimise-call-expression

6.24.1

1.923.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.924 is-number 7.0.0 7.0.0

1.924.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.925 base64-arraybuffer 0.1.4 0.1.4

1.925.1 Available under license:
Copyright (c) 2012 Niklas von Hertzen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.926 jackson 2.10.5**

**1.926.1 Available under license :**

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor annotations is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.927 babel-plugin-transform-es2015-typeof-symbol 6.23.0 6.23.0
1.927.1 Available under license :
   The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.928 is-posix-bracket 0.1.1
1.928.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.929 raw-body 2.4.0 2.4.0

1.929.1 Available under license :
   The MIT License (MIT)

   Copyright (c) 2013-2014 Jonathan Ong <me@jongleberry.com>
   Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.930 fstream 1.0.12 1.0.12

1.930.1 Available under license :
   The ISC License

   Copyright (c) Isaac Z. Schlueter and Contributors

   Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.931 jython 2.5.2

1.931.1 Available under license:

```
# Copyright (c) 1998-2000 John Aycock
#
# Permission is hereby granted, free of charge, to any person obtaining
# a copy of this software and associated documentation files (the
# "Software"), to deal in the Software without restriction, including
# without limitation the rights to use, copy, modify, merge, publish,
# distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so, subject to
# the following conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
# MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
# IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
# CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
# TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
/*
 * Copyright 2000 Finn Bock
 *
 * This program contains material copyrighted by:
 * Copyright (c) 1997-2000 by Secret Labs AB. All rights reserved.
 *
 * This version of the SRE library can be redistributed under CNRI's
 * Python 1.6 license. For any other use, please contact Secret Labs
 * AB (info@pythonware.com).
 *
 * Portions of this engine have been developed in cooperation with
 * CNRI. Hewlett-Packard provided funding for 1.6 integration and
 * other compatibility work.
 */
/*
 * The Apache Software License, Version 1.1
```
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*

The Jython License

==================================================================================
A. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING JYTHON

Python Software Foundation License Version 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Jython") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Jython alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2007 Python Software Foundation; All Rights Reserved" are retained in Jython alone or in any derivative version prepared by Licensee.
3. In the event Licensee prepares a derivative work that is based on or incorporates Jython or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Jython.

4. PSF is making Jython available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF JYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF JYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING JYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Jython, Licensee agrees to be bound by the terms and conditions of this License Agreement.

Jython 2.0, 2.1 License
--------------------------------------------
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the Jython Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

JPython 1.1.x Software License.

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and using JPython version 1.1.x in source or binary form and its associated documentation as provided herein ("Software").

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a non-exclusive, non-transferable, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright 1996-1999 Corporation for National Research Initiatives; All Rights Reserved" are both retained in the Software, alone or in any derivative version prepared by Licensee.

Alternatively, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes), provided, however, that such text is displayed prominently in the Software alone or in any derivative version prepared by Licensee: "JPython (Version 1.1.x) is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1006. The License may also be obtained from a proxy server on the Web using the following URL: http://hdl.handle.net/1895.22/1006."
3. In the event Licensee prepares a derivative work that is based on or incorporates the Software or any part thereof, and wants to make the derivative work available to the public as provided herein, then Licensee hereby agrees to indicate in any such work, in a prominently visible way, the nature of the modifications made to CNRI's Software.

4. Licensee may not use CNRI trademarks or trade name, including JPython or CNRI, in a trademark sense to endorse or promote products or services of Licensee, or any third party. Licensee may use the mark JPython in connection with Licensee's derivative versions that are based on or incorporate the Software, but only in the form "JPython-based ________________,” or equivalent.

5. CNRI is making the Software available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

6. CNRI SHALL NOT BE LIABLE TO LICENSEE OR OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY SO THE ABOVE DISCLAIMER MAY NOT APPLY TO LICENSEE.

7. This License Agreement may be terminated by CNRI (i) immediately upon written notice from CNRI of any material breach by the Licensee, if the nature of the breach is such that it cannot be promptly remedied; or (ii) sixty (60) days following notice from CNRI to Licensee of a material remediable breach, if Licensee has not remedied such breach within that sixty-day period.

8. This License Agreement shall be governed by and interpreted in all respects by the law of the State of Virginia, excluding conflict of law provisions. Nothing in this Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee.

9. By clicking on the "ACCEPT" button where indicated, or by installing, copying or otherwise using the Software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

B. HISTORY OF THE SOFTWARE
JPython was created in late 1997 by Jim Hugunin. Jim was also the primary developer while he was at CNRI. In February 1999 Barry Warsaw took over as primary developer and released JPython version 1.1.

In October 2000 Barry helped move the software to SourceForge where it was renamed to Jython. Jython 2.0 and 2.1 were developed under the Jython specific license below.

From the 2.2 release on, Jython contributors have signed Python Software Foundation contributor agreements and releases are covered under the Python Software Foundation license version 2.

The standard library is covered by the Python Software Foundation license as well. See the Lib/LICENSE file for details.

The zxJDBC package was written by Brian Zimmer and originally licensed under the GNU Public License. The package is now covered by the Jython Software License.

The command line interpreter is covered by the Apache Software License. See the org/apache/LICENSE file for details.

1.932 okhttp 4.4.0
1.932.1 Available under license :
Note that publicsuffixes.gz is compiled from The Public Suffix List: https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0: https://mozilla.org/MPL/2.0/

1.933 source-map-url 0.4.0 0.4.0
1.933.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.934 fresh 0.3.0 0.3.0

1.934.1 Available under license :
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.935 json-schema-validator 0.1.16

1.935.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016 Network New Technologies Inc.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* You may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/TypeFactory.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/JsonValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/PropertiesValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/OneOfValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/MaxLengthValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/RefValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/FormatValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/JsonSchemaException.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/MinPropertiesValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/MultipleOfValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/BaseJsonValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/RequiredValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/UniqueItemsValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/MaxPropertiesValidator.java
* /opt/cola/permits/1001160735_1606872800.56/0/json-schema-validator-0-1-16-sources-jar/com/networknt/schema/ValidatorTypeCode.java
~ You may obtain a copy of the License at
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.

Found in path(s):

1.936 moment-timezone 0.5.21 0.5.21
1.936.1 Available under license :
The MIT License (MIT)

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.937 async-replace-with-limit 1.1.0 1.1.0
1.937.1 Available under license :
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.938 escape-html 1.0.3 1.0.3
1.938.1 Available under license :
(The MIT License)

Copyright (c) 2012-2013 TJ Holowaychuk
Copyright (c) 2015 Andreas Lubbe
Copyright (c) 2015 Tiancheng "Timothy" Gu

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
1.939 orc-shims 1.5.6

1.939.1 Available under license:

ORC Shims
Copyright 2013-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.940 inherits 2.0.3 2.0.3

1.940.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.941 babel-messages 6.23.0
1.941.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE 
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION 
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright 
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright 
  notice, this list of conditions and the following disclaimer in the 
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" 
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY 
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES 
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; 
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND 
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF 
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all 
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.942 httpcore5-h 5.0.2

1.942.1 Available under license:

Apache HttpComponents Core HTTP/2
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.943 lodash-isfunction 3.0.9 3.0.9

1.943.1 Available under license :

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.944 is-fullwidth-code-point 3.0.0

1.944.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
1.945 servlet-api 3.1.0

1.945.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and
(b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has
deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the
Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made
available in Source Code form and that Source Code form must be distributed only under the terms of this License.
You must include a copy of this License with every copy of the Source Code form of the Covered Software You
distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a
medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You
represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the
Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered
Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial
Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the
applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee
for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must
make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone,
and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the
Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or
alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of
this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if
the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise
make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License.
Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.946 locate-path 3.0.0 3.0.0

1.946.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.947 node-forge 0.10.0 0.10.0

1.947.1 Available under license:
You may use the Forge project under the terms of either the BSD License or the GNU General Public License (GPL) Version 2.

The BSD License is recommended for most projects. It is simple and easy to understand and it places almost no restrictions on what you can do with the Forge project.

If the GPL suits your project better you are also free to use Forge under that license.

You don't have to do anything special to choose one license or the other and you don't have to notify anyone which license you are using. You are free to use this project in commercial projects as long as the copyright header is left intact.

If you are a commercial entity and use this set of libraries in your
commercial software then reasonable payment to Digital Bazaar, if you can afford it, is not required but is expected and would be appreciated. If this library saves you time, then it’s saving you money. The cost of developing the Forge software was on the order of several hundred hours and tens of thousands of dollars. We are attempting to strike a balance between helping the development community while not being taken advantage of by lucrative commercial entities for our efforts.

New BSD License (3-clause)
Copyright (c) 2010, Digital Bazaar, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Digital Bazaar, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DIGITAL BAZAAR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free...
Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

1.948 rc 1.2.8

1.948.1 Available under license :

Copyright (c) 2013, Dominic Tarr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
IN THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License, Version 2.0

Copyright (c) 2011 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.949 has-flag 3.0.0 3.0.0

1.949.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.950 find-up 4.1.0

1.950.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.951 socket.io-parser 3.4.1 3.4.1
1.951.1 Available under license :
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.952 de.huxhorn.sulky.io 0.9.17
1.952.1 Available under license :
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this
license document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this
license document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of
this License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version
of the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user’s computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the Library.

1.953 ultron 1.0.2 1.0.2
1.953.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.954 call-me-maybe 1.0.1 1.0.1
1.954.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Eric McCarthy

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.955 cycle 1.0.3 1.0.3
1.955.1 Available under license :
/*
cycle.js
2013-02-19

Public Domain.

NO WARRANTY EXPRESSED OR IMPLIED. USE AT YOUR OWN RISK.

This code should be minified before deployment.
See http://javascript.crockford.com/jsmin.html

USE YOUR OWN COPY. IT IS EXTREMELY UNWISE TO LOAD CODE FROM SERVERS YOU DO NOT CONTROL.
*/

1.956 jsreport-handlebars 2.1.0 2.1.0
1.956.1 Available under license :
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

   b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.957 zip 1.2.0 1.2.0
1.957.1 Available under license:

Copyright 2009, 2010 Kristopher Michael Kowal. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.958 jsreport-cli 2.2.5 2.2.5
1.958.1 Available under license:

MIT License

Copyright (c) 2017 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1.959 commons-lang3 3.10

1.959.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (https://www.apache.org/).

1.960 emojis-list 3.0.0 3.0.0
1.960.1 Available under license:
The MIT License (MIT)

Copyright 2015 Kiko Beats

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.961 phantom-page-eval 1.2.0 1.2.0

1.961.1 Available under license:

The MIT License (MIT)

Copyright (c) 2018 bjrmatos

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.962 backo2 1.0.2 1.0.2

1.962.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.963 moment 2.22.1 2.22.1

1.963.1 Available under license:

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.964 commons-collections 4.4

1.964.1 Available under license:

Apache Commons Collections
Copyright 2001-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.965 jersey-container-servlet 2.25.1

1.965.1 Available under license:
CDDL+GPL License
http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.966 commons-codec 1.12

1.966.1 Available under license:
Apache Commons Codec
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

src/test/org/apache/commons/codec/language[DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.967 jmespath-java 1.11.375

1.967.1 Available under license:

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/META-INF/maven/com.amazonaws/jmespath-java/pom.xml

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/JmesPathLiteral.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/JmesPathSubExpression.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/OpGreaterThan.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/JmesPathFunction.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/JmesPathField.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/Comparator.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/JmesPathAndExpression.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/OpGreaterThanOrEqualTo.java
* /opt/cola/permits/1135892278_1613627038.25/0/jmespath-java-1-11-375-sources-1-jar/com/amazonaws/jmespath/OpGreaterThanOrEqualTo.java
1.968 nanoid 1.0.7 1.0.7

1.968.1 Available under license:

The MIT License (MIT)

Copyright 2017 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the “Software”), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.969 batik-shared-resources 1.13

1.969.1 Available under license :
Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software's
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.970 querystringify 2.2.0 2.2.0
1.970.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.971 swagger-models 2.0.7
1.971.1 Available under license:
/**
 * Copyright 2017 SmartBear Software
 * <p>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package io.swagger.v3.oas.models.info;
import java.util.Objects;

/**
 * License
 *
 * @see "https://github.com/OAI/OpenAPI-Specification/blob/3.0.1/versions/3.0.1.md/licenseObject"
 */

public class License {
    private String name = null;
    private String url = null;
    private java.util.Map<String, Object> extensions = null;

    /**
     * returns the name property from a License instance.
     *
     * @return String name
     **/
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public License name(String name) {
        this.name = name;
        return this;
    }

    /**
     * returns the url property from a License instance.
     *
     * @return String url
     **/
    public String getUrl() {
        return url;
    }
    public void setUrl(String url) {
        this.url = url;
    }
    public License url(String url) {
        this.url = url;
        return this;
    }
}
return this;
}

@Override
public boolean equals(java.lang.Object o) {
    if (this == o) {
        return true;
    }
    if (o == null || getClass() != o.getClass()) {
        return false;
    }
    License license = (License) o;
    return Objects.equals(this.name, license.name) &&
            Objects.equals(this.url, license.url) &&
            Objects.equals(this.extensions, license.extensions);
}

@Override
public int hashCode() {
    return Objects.hash(name, url, extensions);
}

public java.util.Map<String, Object> getExtensions() {
    return extensions;
}

public void addExtension(String name, Object value) {
    if (name == null || name.isEmpty() || !name.startsWith("x-")) {
        return;
    }
    if (this.extensions == null) {
        this.extensions = new java.util.LinkedHashMap<>();
    }
    this.extensions.put(name, value);
}

public void setExtensions(java.util.Map<String, Object> extensions) {
    this.extensions = extensions;
}

public License extensions(java.util.Map<String, Object> extensions) {
    this.extensions = extensions;
    return this;
}

@Override
public String toString() {
    StringBuilder sb = new StringBuilder();
sb.append("class License {\n");
    sb.append("    name: ").append(toIndentedString(name)).append("\n");
    sb.append("    url: ").append(toIndentedString(url)).append("\n");
    sb.append("}\n");
    return sb.toString();
}
/**
 * Convert the given object to string with each line indented by 4 spaces
 * (except the first line).
 */
private String toIndentedString(java.lang.Object o) {
    if (o == null) {
        return "null";
    }
    return o.toString().replace("\n", "\n   ");
}

1.972 deep-equal 0.2.2 0.2.2

1.972.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.973 spss-reader 1.3
1.973.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file
* except in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
* ANY KIND, either express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/LongStringRecord.java

No license file was found, but licenses were detected in source scan.

/*
 *
* Copyright 2017 BeDataDriven Groep BV
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file
* except in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
* ANY KIND, either express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/MissingValuesHeader.java
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/FileHeader.java
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/LongVariableNamesRecord.java
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/ValueLabelsRecord.java
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/CaseBuffer.java
* /opt/cola/permits/1202447241_1631568244.43/0/spss-reader-1-3-sources-jar/com/bedatadriven/spss/DocumentRecord.java
1.974 kind-of 5.1.0 5.1.0

1.974.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.975 guava 28.2-jre

1.975.1 Available under license:
Apache License

Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.976 balanced-match 1.0.0 1.0.0

1.976.1 Available under license :

(MIT)

Copyright (c) 2013 Julian Gruber &lt;julian@juliangruber.com&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.977 lodash.isboolean 3.0.3 3.0.3

1.977.1 Available under license:

Copyright 2012-2016 The Dojo Foundation <http://dojofoundation.org/>
Based on Underscore.js, copyright 2009-2016 Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.978 filter-obj 1.1.0 1.1.0

1.978.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.979 emoji-regex 8.0.0

1.979.1 Available under license:
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.980 babel-helper-compilation-targets 7.16.3

1.980.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.981 istanbul-lib-hook 3.0.0

1.981.1 Available under license : 
Copyright 2012-2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the
names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.982 merge2 1.3.0 1.3.0

1.982.1 Available under license : 
The MIT License (MIT)

Copyright (c) 2014-2019 Teambition

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.983 metadata-extractor 2.13.0

1.983.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2002-2019 Drew Noakes and contributors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.984 batik-extension 1.13

1.984.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for Standardization for the definition of character entities used in the software's documentation.
This product includes images from the Tango Desktop Project (http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme (http://www.jesusda.com/projects/pasodoble).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.985 jetty-util 9.4.34.v20201102
1.985.1 Available under license:

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS
A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a
court requires any other Contributor to pay any damages as a result, the
Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using
and distributing the Program and assumes all risks associated with its
exercise of rights under this Agreement, including but not limited to the
risks and costs of program errors, compliance with applicable laws, damage to
or loss of data, programs or equipment, and unavailability or interruption of
operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION
LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of the
remainder of the terms of this Agreement, and without further action by the
parties hereto, such provision shall be reformed to the minimum extent
necessary to make such provision valid and enforceable.
If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

=================================================================================================

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

=================================================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.986 boost 1.58.0+dfsg 5ubuntu3.1

1.986.1 Available under license :

////////////////////////////////////////////////////////////////////////
//
// (C) Copyright Ion Gaztanaga 2006-2014
// (C) Copyright Microsoft Corporation 2014
//
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE_1_0.txt or copy at
//   http://www.boost.org/LICENSE_1_0.txt)
//
//
////////////////////////////////////////////////////////////////////////

#ifndef BOOST_INTRUSIVE_DETAIL_MPL_HPP
#define BOOST_INTRUSIVE_DETAIL_MPL_HPP

#ifndef BOOST_CONFIG_HPP
  #include <boost/config.hpp>
#endif

#ifndef BOOST_INTRUSIVE_DETAIL_MPL_HPP
#define BOOST_INTRUSIVE_DETAIL_MPL_HPP

#ifndef BOOST_CONFIG_HPP
  # include <boost/config.hpp>
  #endif

```haskell```
```
#if defined(BOOST_HAS_PRAGMA_ONCE)
#pragma once
#endif
#include <boost/intrusive/detail/config_begin.hpp>
#include <cstddef>

namespace boost {
    namespace intrusive {
        namespace detail {

            template <typename T, typename U>
            struct is_same {
                static const bool value = false;
            };

            template <typename T>
            struct is_same<T, T> {
                static const bool value = true;
            };

            template<typename T>
            struct add_const {  typedef const T type;   };

            template<typename T>
            struct remove_const {  typedef  T type;   };

            template<typename T>
            struct remove_const<const T> {  typedef T type;   };

            template<typename T>
            struct remove_cv {  typedef  T type;   };

            template<typename T>
            struct remove_cv<const T> {  typedef T type;   };

            template<typename T>
            struct remove_cv<const volatile T> {  typedef T type;   };

        } // namespace detail
    } // namespace intrusive
} // namespace boost
template<typename T>
struct remove_cv<volatile T>
{
    typedef T type;
};

template<class T>
struct remove_reference
{
    typedef T type;
};

template<class T>
struct remove_reference<T&>
{
    typedef T type;
};

template<class T>
struct remove_pointer
{
    typedef T type;
};

template<class T>
struct remove_pointer<T*>
{
    typedef T type;
};

template<class T>
struct add_pointer
{
    typedef T *type;
};

typedef char one;
struct two {one _[2];};

template< bool C_ >
struct bool_
{
    static const bool value = C_;
};

template< class Integer, Integer Value >
struct integer
{
    static const Integer value = Value;
};
typedef bool_<true> true_;
typedef bool_<false> false_;
typedef true_ true_type;
typedef false_ false_type;
typedef char yes_type;
struct no_type
{
    char padding[8];
};
template <bool B, class T = void>
struct enable_if_c {
    typedef T type;
};
template <class T>
struct enable_if_c<false, T> {};
template <class Cond, class T = void>
struct enable_if : public enable_if_c<Cond::value, T> {};
template<class F, class Param>
struct apply
{
    typedef typename F::template apply<Param>::type type;
};
#if defined(_MSC_VER) && (_MSC_VER >= 1400)
template <class T, class U>
struct is_convertible
{
    static const bool value = __is_convertible_to(T, U);
};
#else

template <class T, class U>
class is_convertible
{
    typedef char true_t;
    class false_t { char dummy[2]; }; //use any_conversion as first parameter since in MSVC
    //overaligned types can't go through ellipsis
    static false_t dispatch(...);
};
static true_t dispatch(U);
static typename remove_reference<T>::type &trigger();
public:
static const bool value = sizeof(dispatch(trigger())) == sizeof(true_t);
};

#endif

template<
    bool C
, typename T1
, typename T2
>
struct if_c
{
    typedef T1 type;
};

template<
    typename T1
, typename T2
>
struct if_c<false,T1,T2>
{
    typedef T2 type;
};

template<
    typename C
, typename T1
, typename T2
>
struct if_
{
    typedef typename if_c<0 != C::value, T1, T2>::type type;
};

template<
    bool C
, typename F1
, typename F2
>
struct eval_if_c
    : if_c<C,F1,F2>::type
{
};

template<
    typename C

struct eval_if
    : if_<C,T1,T2>::type
{};

// identity is an extension: it is not part of the standard.

template <class T>
struct identity
{
    typedef T type;
};

template<class T, bool Add>
struct add_const_if_c
{
    typedef typename if_c
        < Add
        , typename add_const<T>::type
        , T
        >::type type;
};

//boost::alignment_of yields to 10K lines of preprocessed code, so we
//need an alternative

template <typename T> struct alignment_of;

template <typename T>
struct alignment_of_hack
{
    char c;
    T t;
    alignment_of_hack();
};

template <unsigned A, unsigned S>
struct alignment_logic
{
    static const std::size_t value = A < S ? A : S;
};

template< typename T >
struct alignment_of
{
    static const std::size_t value = alignment_logic
        < sizeof(alignment_of_hack<T>) - sizeof(T)
template<class Class>
class is_empty_class
{
    template <typename T>
    struct empty_helper_t1 : public T
    {
        empty_helper_t1();
        int i[256];
    };

    struct empty_helper_t2
    { int i[256]; };

public:
    static const bool value = sizeof(empty_helper_t1<Class>) == sizeof(empty_helper_t2);
};

template<std::size_t S>
struct ls_zeros
{
    static const std::size_t value = (S & std::size_t(1)) ? 0 : (1 + ls_zeros<(S>>1u)>::value);
};

template<>
struct ls_zeros<0>
{
    static const std::size_t value = 0;
};

template<>
struct ls_zeros<1>
{
    static const std::size_t value = 0;
};

template <typename T> struct unvoid_ref { typedef T &type; };
template <> struct unvoid_ref<void> { struct type_impl { }; typedef type_impl & type; };
template <> struct unvoid_ref<const void> { struct type_impl { }; typedef type_impl & type; };

// Infrastructure for providing a default type for T::TNAME if absent.
#define BOOST_INTRUSIVE_INSTANTIATE_DEFAULT_TYPE_TMPLT(TNAME)     
template <typename T, typename DefaultType>                    
struct boost_intrusive_default_type_ ## TNAME                  
{                                                              


template <typename X>
    \ /
static char test(int, typename X::TNAME*);
    \ /

template <typename X>
    \ /
static int test(...);
    \ /

struct DefaultWrap { typedef DefaultType TNAME; }; \ /

static const bool value = (1 == sizeof(test<T>(0, 0)));

typedef typename
    ::boost::intrusive::detail::if_c
    \ <value, T, DefaultWrap>::type::TNAME type;
}; \ /
//

#define BOOST_INTRUSIVE_OBTAIN_TYPE_WITH_DEFAULT(INSTANTIATION_NS_PREFIX, T, TNAME, TIMPL) \
    typename INSTANTIATION_NS_PREFIX
    boost_intrusive_default_type_ ## TNAME< T, TIMPL >::type
    \ /
//

#define BOOST_INTRUSIVE_INSTANTIATE_EVAL_DEFAULT_TYPE_TMPLT(TNAME)
    template <typename T, typename DefaultType>
    struct boost_intrusive_eval_default_type_ ## TNAME
    { \ /
        template <typename X>
        \ /
static char test(int, typename X::TNAME*);
        \ /

        template <typename X>
        \ /
static int test(...);
        \ /

        struct DefaultWrap \ /
        { typedef typename DefaultType::type TNAME; }; \ /

        static const bool value = (1 == sizeof(test<T>(0, 0)));

        typedef typename
            ::boost::intrusive::detail::eval_if_c
            \ < value
            , ::boost::intrusive::detail::identity<T>
            , ::boost::intrusive::detail::identity<DefaultWrap>
            >::type::TNAME type;
    }; \ /
//

#define BOOST_INTRUSIVE_OBTAIN_TYPE_WITH_EVAL_DEFAULT(INSTANTIATION_NS_PREFIX, T, TNAME, TIMPL) \
#define BOOST_INTRUSIVE_INTERNAL_STATIC_BOOL_IS_TRUE(TRAITS_PREFIX, TYPEDEF_TO_FIND) \
struct TRAITS_PREFIX##_bool
{
    template<bool Add>
    struct two_or_three {one _(2 + Add);};
    template <class U> static one test(...);
    template <class U> static two_or_three<U::TYPEDEF_TO_FIND> test (int);
    static const std::size_t value = sizeof(test<T>(0));
};

struct TRAITS_PREFIX##_bool_is_true
{
    static const bool value = TRAITS_PREFIX##_bool<T>::value > sizeof(one)*2;
};

#define BOOST_INTRUSIVE_HAS_STATIC_MEMBER_FUNC_SIGNATURE(TRAITS_NAME, FUNC_NAME) \
    template <typename U, typename Signature> \
    class TRAITS_NAME \
    {
    private:
        template<Signature> struct helper;
        template<typename T> \
        static ::boost::intrusive::detail::yes_type check(helper<&T::FUNC_NAME>*); \
    public:
        static const bool value = sizeof(check<U>(0)) == sizeof(::boost::intrusive::detail::yes_type);
    };

#define BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED(TRAITS_NAME, FUNC_NAME) \
    template <typename Type> \
    struct TRAITS_NAME \
    {
        struct BaseMixin \
        {
            void FUNC_NAME();
        };
        struct Base : public Type, public BaseMixin { Base(); };
    };
}
template <typename U> \
static ::boost::intrusive::detail::no_type check(U*, Helper<void (BaseMixin::*)(), &U::FUNC_NAME>* = 0);  
static ::boost::intrusive::detail::yes_type check(...);  
static const bool value = sizeof(::boost::intrusive::detail::yes_type) == sizeof(check((Base*)(0)));  
}  
//
#define BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED_IGNORE_SIGNATURE(TRAITS_NAME, FUNC_NAME) \
BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED(TRAITS_NAME##_ignore_signature, FUNC_NAME)  \
\  
template <typename Type, class> \
struct TRAITS_NAME \
  : public TRAITS_NAME##_ignore_signature<Type> 
{}  
//

template <typename T> 
inline T* addressof(T& obj) 
{
  return static_cast<T*>(
    (static_cast<void*>(
      (const_cast<char*>(
        reinterpret_cast<const char&>(obj))
      )
    ));
  
}  

} //namespace detail 
} //namespace intrusive 
} //namespace boost 

#include <boost/intrusive/detail/config_end.hpp> 

#endif //BOOST_INTRUSIVE_DETAIL_MPL_HPP
All the value based traits in this library conform to MPL’s requirements for
an integral constant type that includes a number of rather intrusive workarounds
for broken compilers.

Purely as an implementation detail, this means that <code class="computeroutput">true_type</code> inherits from <code class="computeroutput">boost</code>, <code class="computeroutput">false_type</code> inherits from <code class="computeroutput">mpl</code>, and <code class="computeroutput">integral_constant</code> inherits from <code class="computeroutput">mpl</code> (provided <code class="computeroutput">T</code> is not <code class="computeroutput">mpl</code>).
```python
#!/usr/bin/env python
from subprocess import check_output as run
from datetime import datetime
from itertools import groupby
from operator import itemgetter
import re
import magic

def authors(filename):
    log = run(['git', 'log', '--follow',
               '--date=short', '--format=%aN%x09%ad', filename],
              universal_newlines=True)
    for line in log.splitlines():
        author, date = line.split('	')
        if author != 'fix-copyright.py':
            yield author, datetime.strptime(date, '%Y-%m-%d')

def new_copyright(filename, previous):
    au = list(authors(filename))
    alldates = map(itemgetter(1), au)
    aup = sorted(au + map(lambda a: (a, None), previous), key=itemgetter(0))
```

---

Open Source Used In DNAC Platform Arica 3316

Frederic Bron, Steve Cleary, Beman Dawes, Aleksey Gurtovoy, Howard Hinnant, Jesse Jones, Mat Marcus, Itay Maman, John Maddock, Alexander Nasonov, Thorsten Ottosen, Roman Perepelitsa, Robert Ramey, Jeremy Siek, Robert Stewart and Steven Watanabe

Distributed under the Boost Software License, Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at <a href="http://www.boost.org/LICENSE_1_0.txt">http://www.boost.org/LICENSE_1_0.txt</a>)

"""
for author, records in groupby(aup, itemgetter(0)):
    dates = filter(None, map(itemgetter(1), records))
    if not dates: dates = alldates
    start = min(dates)
    end = max(dates)
    fmt = '{0}' if start.year == end.year else '{0}-{1}'
    line = 'Copyright ' + fmt.format(start.year, end.year) + ' ' + author
    key = (start, author)
    yield key, line
return map(itemgetter(1), sorted(f()))

def fix_copyright(filename):
    # Find copyright block in original file
    prefix = set()
    names = []
    lines = []
    with open(filename, 'r') as f:
        content = list(f)
        for i, line in enumerate(content[:15]):
            m = re.match(r'^\W*(\(c\))?\scopyright\s*(\(c\))?\s+\d{4}(\s+-\s+\d{4})?\s+(\S*>)\s*\S*,$', line, re.IGNORECASE)
            if m:
                d = m.groupdict()
                prefix.add(d['prefix'])
                lines.append(i)
                names.append(d['name'].strip())
        if len(prefix) != 1:
            print 'Not found:', filename
            return
        prefix = list(prefix)[0]

    print filename
    new = iter(new_copyright(filename, names))
    with open(filename, 'w') as f:
        for i, line in enumerate(content):
            if i in lines:
                for repl in new:
                    print >>f, prefix + repl
            else:
                print >>f, line,
                pass

    def all_files():
        ls = run(['git', 'ls-files'], universal_newlines=True)
        for filename in ls.splitlines():
            if magic.from_file(filename, mime=True).split('/')[0] == 'text':
                yield filename
for f in all_files():
    fix_copyright(f)
[article Invalid copyright
[quickbook 1.5]
[copyright No year]
]

Maybe this should pass?
</section id="date_time.license">
<title>Boost Date-Time Library License</title>
<bridgehead renderas="sect2">Boost Date-Time Library License</bridgehead>
<!--hr /-->
<para>
The following is the overall license for the boost date_time library. This notice is found in all source files related to the library.
</para>
<para>
Copyright © 2002 CrystalClear Software, Inc.
</para>
<para>
Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. CrystalClear Software makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.
</para>
<!--hr /-->
<ulink url="mailto:jeff@crystalclearsoftware.com">Jeff Garland</ulink>&#169; 2000-2002
</section>
#ifndef BOOST_CONTAINER_CONTAINER_DETAIL_MPL_HPP
#define BOOST_CONTAINER_CONTAINER_DETAIL_MPL_HPP

#if defined(BOOST_CONFIG_HPP)
  # include <boost/config.hpp>
#endif

#if defined(BOOST_HAS_PRAGMA_ONCE)
  #pragma once
#endif

#include <boost/container/detail/config_begin.hpp>
#include <boost/container/detail/workaround.hpp>
#include <cstdlib>

namespace boost {
 namespace container {
 namespace container_detail {

 template <class T, T val>
 struct integral_constant
 {
  static const T value = val;
  typedef integral_constant<T,val> type;
 };

 template< bool C_> 
 struct bool_ : integral_constant<bool, C_>
 {
  static const bool value = C_; 
  operator bool() const { return bool_.::value; } 
 };

 template< unsigned V_> 
 struct unsigned_ : integral_constant<unsigned, V_>
{
{  
    static const unsigned value = V_;  
    operator unsigned() const { return unsigned_::value; }  
};

typedef bool_<true>     true_;  
typedef bool_<false>    false_;  

typedef true_   true_type;  
typedef false_  false_type;  

typedef char yes_type;  
struct no_type  
{  
    char padding[8];  
};

template <bool B, class T = void>  
struct enable_if_c  
{  
    typedef T type;  
};

template <class T>  
struct enable_if_c<false, T>  
{};

template <class Cond, class T = void>  
struct enable_if : public enable_if_c<Cond::value, T>  
{};

template <class Cond, class T = void>  
struct disable_if : public enable_if_c<!Cond::value, T>  
{};

template <bool B, class T = void>  
struct disable_if_c : public enable_if_c<!B, T>  
{};
#if defined(_MSC_VER) && (_MSC_VER >= 1400)
    template <class T, class U>  
    struct is_convertible  
    {  
        static const bool value = __is_convertible_to(T, U);  
    };
#else
    template <class T, class U>  
    class is_convertible  
    {  
        typedef char true_t;

class false_t { char dummy[2]; }
//use any_conversion as first parameter since in MSVC
//overaligned types can't go through ellipsis
static false_t dispatch(...);
static true_t  dispatch(U);
static T &trigger();
public:
    static const bool value = sizeof(dispatch(trigger())) == sizeof(true_t);
};

#endif

template<
    bool C
 , typename T1
 , typename T2
 >
struct if_c
{
    typedef T1 type;
};

template<
    typename T1
 , typename T2
 >
struct if_c<false,T1,T2>
{
    typedef T2 type;
};

template<
    typename T1
 , typename T2
 , typename T3
 >
struct if_<
{
    typedef typename if_c<0 != T1::value, T2, T3>::type type;
};

template <class Pair>
struct select1st
{
    typedef Pair argument_type;
    typedef typename Pair::first_type result_type;
template<class OtherPair>
const typename Pair::first_type& operator()(const OtherPair& x) const
{
    return x.first;
}

const typename Pair::first_type& operator()(const typename Pair::first_type& x) const
{
    return x;
}

// identity is an extension: it is not part of the standard.
template <class T>
struct identity
{
    typedef T argument_type;
    typedef T result_type;

    typedef T type;
    const T& operator()(const T& x) const
    {
        return x;
    }
};

template<std::size_t S>
struct ls_zeros
{
    static const std::size_t value = (S & std::size_t(1)) ? 0 : (1u + ls_zeros<(S >> 1u)>::value);
};

template<>
struct ls_zeros<0>
{
    static const std::size_t value = 0;
};

template<>
struct ls_zeros<1>
{
    static const std::size_t value = 0;
};

template <std::size_t OrigSize, std::size_t RoundTo>
struct ct_rounded_size
{
    static const std::size_t value = ((OrigSize-1)/RoundTo+1)*RoundTo;
};

template <typename T> struct unvoid { typedef T type; };
template <> struct unvoid<void> { struct type { }; };
template <> struct unvoid<const void> { struct type { }; };

Open Source Used In DNAC Platform Arica 3322
This is an attempt to acknowledge early contributions to the garbage collector. Later contributions should instead be mentioned in README.changes.

HISTORY -

Early versions of this collector were developed as a part of research projects supported in part by the National Science Foundation and the Defense Advance Research Projects Agency.

The garbage collector originated as part of the run-time system for the Russell programming language implementation. The first version of the garbage collector was written primarily by Al Demers. It was then refined and mostly rewritten, primarily by Hans-J. Boehm, at Cornell U., the University of Washington, Rice University (where it was first used for C and assembly code), Xerox PARC, SGI, and HP Labs. However, significant contributions have also been made by many others.

Some other contributors:

More recent contributors are mentioned in the modification history in README.changes. My apologies for any omissions.

The SPARC specific code was originally contributed by Mark Weiser. The Encore Multimax modifications were supplied by Kevin Kenny (kenny@m.cs.uiuc.edu). The adaptation to the IBM PC/RT is largely due to Vernon Lee, on machines made available to Rice by IBM. Much of the HP specific code and a number of good suggestions for improving the generic code are due to Walter Underwood. Robert Brazile (brazile@diamond.bbn.com) originally supplied the ULTRIX code. Al Dosser (dossier@src.dec.com) and Regis Cridlig (Regis.Cridlig@cl.cam.ac.uk) subsequently provided updates and information on variation between ULTRIX systems. Parag Patel (parag@netcom.com) supplied the A/UX code. Jesper Peterson (jep@mtiame.mtia.oz.au), Michel Schinz, and Martin Tauchmann (martintauchmann@bigfoot.com) supplied the Amiga port. Thomas Funke (thf@zelator.in-berlin.de(?)) and Brian D.Carlstrom (bdc@clark.lcs.mit.edu) supplied the NeXT ports. Douglas Steel (doug@wg.icl.co.uk) provided ICL DRS6000 code. Bill Janssen (janssen@parc.xerox.com) supplied the SunOS dynamic loader specific code. Manuel Serrano (serrano@cornas.inria.fr) supplied linux and Sony News specific code. Al Dosser provided Alpha/OSF/1 code. He and
Dave Detlefs(detlefs@src.dec.com) also provided several generic bug fixes.
Alistair G. Crooks(acg@uts.amdahl.com) supplied the NetBSD and 386BSD ports.
Jeffrey Hsu (hsu@soda.berkeley.edu) provided the FreeBSD port.
Brent Benson (brent@jade.ssd.csd.harris.com) ported the collector to
a Motorola 88K processor running CX/UX (Harris NightHawk).
Ari Huttunen (Ari.Huttunen@hut.fi) generalized the OS/2 port to
nonIBM development environments (a nontrivial task).
Patrick Beard (beard@cs.ucdavis.edu) provided the initial MacOS port.
David Chase, then at Olivetti Research, suggested several improvements.
Scott Schwartz (schwartz@groucho.cse.psu.edu) supplied some of the
code to save and print call stacks for leak detection on a SPARC.
Jesse Hull and John Ellis supplied the C++ interface code.
Zhong Shao performed much of the experimentation that led to the
current typed allocation facility. (His dynamic type inference code hasn’t
made it into the released version of the collector, yet.)

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the “Software”) to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS“, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

Software License, Version 1.0

Copyright 2002-2003, Trustees of Indiana University.
Copyright 2000-2001, University of Notre Dame.
All rights reserved.

Indiana University has the exclusive rights to license this product under the
following license.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* All redistributions of source code must retain the above copyright notice, the list of authors in the original source code, this list of conditions and the disclaimer listed in this license;

* All redistributions in binary form must reproduce the above copyright notice, this list of conditions and the disclaimer listed in this license in the documentation and/or other materials provided with the distribution;

* Any documentation included with all redistributions must include the following acknowledgement:

  "This product includes software developed at the University of Notre Dame and the Pervasive Technology Labs at Indiana University. For technical information contact Andrew Lumsdaine at the Pervasive Technology Labs at Indiana University. For administrative and license questions contact the Advanced Research and Technology Institute at 351 West 10th Street. Indianapolis, Indiana 46202, phone 317-278-4100, fax 317-274-5902."

Alternatively, this acknowledgement may appear in the software itself, and wherever such third-party acknowledgments normally appear.

* The name Indiana University, the University of Notre Dame or "Caramel" shall not be used to endorse or promote products derived from this software without prior written permission from Indiana University. For written permission, please contact Indiana University Advanced Research & Technology Institute.

* Products derived from this software may not be called "Caramel", nor may Indiana University, the University of Notre Dame or "Caramel" appear in their name, without prior written permission of Indiana University Advanced Research & Technology Institute.

Indiana University provides no reassurances that the source code provided does not infringe the patent or any other intellectual property rights of any other entity. Indiana University disclaims any liability to any recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise.

LICENSEE UNDERSTANDS THAT SOFTWARE IS PROVIDED "AS IS" FOR WHICH NO WARRANTIES AS TO CAPABILITIES OR ACCURACY ARE MADE. INDIANA UNIVERSITY GIVES NO WARRANTIES AND MAKES NO REPRESENTATION THAT SOFTWARE IS FREE OF INFRINGEMENT OF THIRD PARTY PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS. INDIANA UNIVERSITY MAKES NO WARRANTIES THAT SOFTWARE IS FREE FROM "BUGS", "VIRUSES", "TROJAN HORSES", "TRAP DOORS", "WORMS", OR OTHER HARMFUL CODE. LICENSEE ASSUMES THE ENTIRE RISK AS TO THE PERFORMANCE OF SOFTWARE AND/OR ASSOCIATED MATERIALS, AND TO THE PERFORMANCE AND VALIDITY OF INFORMATION GENERATED USING SOFTWARE.
The following people hereby grant permission to replace all existing licenses on their contributions to Boost with the Boost Software License, Version 1.0. (boostinspect:nolicense boostinspect:nocopyright)

Aleksey Gurtovoy (agurtovoy@meta-comm.com)
Andrei Alexandrescu (andrewalex - at - hotmail.com) (See Boost list message of August 12, 2004 11:06:58 AM EST)
Andrew Lumsdaine()
Anthony Williams (anthony -at- justsoftwaresolutions.co.uk)
Beman Dawes (bdawes@acm.org)
Brad King (brad.king -at- kitware.com) (See Boost list message of Wed, 21 Jul 2004 11:15:46 -0400)
Brian Osman (osman -at- vvisions.com) (See CVS log)
Bruce Barr (schmoost -at- yahoo.com) (See Boost list of Mon, 16 Aug 2004 15:06:43 -0500)
Bruno da Silva de Oliveira (bruno - at - esss.com.br)
Christain Engstrom (christian.engstrom -at- glindra.org) (See Boost list message of Mon, 30 Aug 2004 14:31:49 +0200)
Cromwell D Enage (sponage -at- yahoo.com) (See Boost list message of August 12, 2004 11:49:13 AM EST)
Dan Gohman (dgj -at- cray.com) (See Boost list messsage of Sat, 21 Aug 2004 10:54:59 +0100)
Dan Nuffer (dan -at- nuffer.name)
Daniel Frey (d.frey -at- gmx.de, daniel.frey -at- aixigo.de)
Daniel Nuffer (dan -at- nuffer.name)
Darin Adler (darin -at- bentspoon.com) (Email to Andreas Huber, see change log)
Daryle Walker (daryle-w - at - hotmail.com)
Dave Abrahams (dave@boost-consulting.com)
Dave Moore (dmoore -at- viefinancial.com) (See Boost list message of 18 Dec 2003 15:35:50 -0500)
David Abrahams (dave@boost-consulting.com)
Dietmar Kuehl (dietmar_kuehl -at- yahoo.com) (Email to Andreas Huber, see change log)
Douglas Gregor (gregod -at- cs.rpi.edu, dgregor -at- cs.indiana.edu, doug.gregor -at- gmail.com)
Dr John Maddock (john - at - johnmaddock.co.uk)
Edward D. Brey (brey -at- ductape.net) (Email to Andreas Huber, see change log)
Eric Ford (un506n902 -at- sneakemail.com) (See Boost list message of Sun, 15 Aug 2004 10:29:13 +0100)
Eric Friedman (ebf@users.sourceforge.net)
Eric Niebler (eric@boost-consulting.com)
Fernando Cacciola (fernando_cacciola@ciudad.com.ar)
Fernando Luis Cacciola Carballal (fernando_cacciola@ciudad.com.ar)
Francois Faure (Francois.Faure -at- imag.fr) (See CVS log)
Gary Powell (powellg - at - amazon.com) (See Boost list message of 10 Feb 2004 14:22:46 -0800)
Gennadiy Rozental (rogeeff -at- mail.com) (Email to Andreas Huber, see change log)
Gottfried Ganssauge (Gottfried.Ganssauge -at- HAUFE.DE) (See Boost List message of Mon, 16 Aug 2004 10:09:19 +0200)
Gottfried Ganssauge (Gottfried.Ganssauge -at- HAUFE.DE) (Alternative spelling of Gottfried Ganssauge)
Greg Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)
Gregory Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)
Gunter Winkler (gunter.winkler -at- unibw-muenchen.de) (See Boost List message of Mon, 16 Aug 2004 10:24:17 +0200)
Hartmut Kaiser (hartmut.kaiser -at- gamil.com)
Herve Bronnimann (hbr -at- poly.edu)
Herv Brnnimann (hbr -at- poly.edu)
Use of this software is granted under one of the following two licenses, to be chosen freely by the user.


Using this software is granted under the Boost Software License. The license text is as follows:

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the “Software”) to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2. The MIT License

Using this software is granted under the MIT License. The license text is as follows:

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

===========================================================================

Copyright (c) 2007-2011 Barend Gehrels, Amsterdam, the Netherlands.
Copyright (c) 2008-2011 Bruno Lalande, Paris, France.
Copyright (c) 2009-2011 Mateusz Loskot, London, UK.

Use, modification and distribution is subject to the Boost Software License,
Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at
http://www.boost.org/LICENSE_1_0.txt)
===========================================================================

The default copyright note for C++ source files reads:

// Boost.Geometry (aka GGL, Generic Geometry Library)

// Copyright (c) 2007-2011 Barend Gehrels, Amsterdam, the Netherlands.
// Copyright (c) 2008-2011 Bruno Lalande, Paris, France.
// Copyright (c) 2009-2011 Mateusz Loskot, London, UK.

// Parts of Boost.Geometry are redesigned from Geodan's Geographic Library
// (geolib/GGL), copyright (c) 1995-2010 Geodan, Amsterdam, the Netherlands.

// Use, modification and distribution is subject to the Boost Software License,
// Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at
// http://www.boost.org/LICENSE_1_0.txt)

Exceptions:
1) Major work of any author -> might change order, change date
2) Exclusive work of one author including design -> might take sole copyright
Examples:
- adapted geometry types e.g. array, Boost.Polygon
- implemented file-format e.g. WKB
- implemented specific strategy or algorithm e.g. intersections
3) Examples -> might take sole copyright
4) Tests -> might take sole copyright
5) Docs -> might take sole copyright
6) Utilities e.g. converters -> might take sole copyright

The copyright note for other (source) files as .py, Jamfiles, etc is similar and comments are changed accordingly.

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE article PUBLIC "-//Boost//DTD BoostBook XML V1.0//EN" "http://www.boost.org/tools/boostbook/dtd/boostbook.dtd">
<article id="copyright_test" last-revision="DEBUG MODE Date: 2000/12/20 12:00:00 $" xmlns:xi="http://www.w3.org/2001/XInclude">
<title>Copyright Test</title>
<articleinfo>
<copyright>
<year>1963</year> <year>1964</year> <year>1965</year> <holder>Jane Doe</holder>
</copyright>
<copyright>
<year>2018</year> <holder>Joe Blow, John Coe</holder>
</copyright>
<copyright>
<year>1977</year> <year>1985</year> <holder>someone else</holder>
</copyright>
</articleinfo>
</article>
[article Invalid copyright years
[quickbook 1.5]
[copyright 2010-2001 Timequake man]
]
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstart-Name: boost
Source: http://sourceforge.net/projects/boost/files/boost
Files-Excluded: doc/images/valid-html401.png

Files: *
Copyright:
License: Boost
This software is a collection of libraries from the Boost.org site.
Most of the libraries use the Boost Software License 1.0, which reads as follows.
Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

There are a few files under a license different from the Boost Software License; however, all licenses do follow these guidelines:

License requirements

Must be simple to read and understand.

Must grant permission to copy, use and modify the software for any use (commercial and non-commercial) for no fee.

Must require that the license appear on all copies of the software source code.

Must not require that the license appear with executables or other binary uses of the library.

Must not require that the source code be available for execution or other binary uses of the library.

May restrict the use of the name and description of the library to the standard version found on the Boost web site.

Files: tools/jam
License:
/*
Copyright 1993-2002 Christopher Seiwald and Perforce Software, Inc.

License is hereby granted to use this software and distribute it freely, as long as this copyright notice is retained and modifications are clearly marked.

ALL WARRANTIES ARE HEREBY DISCLAIMED.

This file is ALSO:

* Copyright 2001-2004 David Abrahams.
* Distributed under the Boost Software License, Version 1.0.
* (See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)

* Copyright 1988, 1989 Hans-J. Boehm, Alan J. Demers
* Copyright (c) 1991-1994 by Xerox Corporation. All rights reserved.
* Copyright (c) 1998-1999 by Silicon Graphics. All rights reserved.
* Copyright (c) 1999 by Hewlett-Packard Company. All rights reserved.

THIS MATERIAL IS PROVIDED AS IS, WITH ABSOLUTELY NO WARRANTY EXPRESSED OR IMPLIED. ANY USE IS AT YOUR OWN RISK.

Permission is hereby granted to use or copy this program for any purpose, provided the above notices are retained on all copies.
Permission to modify the code and to distribute modified code is granted, provided the above notices are retained, and a notice that the code was modified is included with the above copyright notice.

Files: boostbook/xsl/caramel
License: Software License, Version 1.0

Copyright 2002-2003, Trustees of Indiana University.
Copyright 2000-2001, University of Notre Dame.
All rights reserved.

Indiana University has the exclusive rights to license this product under the following license.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* All Redistributions of source code must retain the above copyright notice,
  the list of authors in the original source code, this list of conditions
  and the disclaimer listed in this license;

* All Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the disclaimer listed in this license
  in the documentation and/or other materials provided with the distribution;

* Any documentation included with all Redistributions must include the
  following acknowledgement:

  "This product includes software developed at the University of Notre Dame
  and the Pervasive Technology Labs at Indiana University. For technical
  information contact Andrew Lumsdaine at the Pervasive Technology Labs at
  Indiana University. For administrative and license questions contact the
  Advanced Research and Technology Institute at 351 West 10th Street.
  Indianapolis, Indiana 46202, phone 317-278-4100, fax 317-274-5902."

Alternatively, this acknowledgement may appear in the software itself, and
wherever such third-party acknowledgments normally appear.

* The name Indiana University, the University of Notre Dame or "Caramel"
  shall not be used to endorse or promote products derived from this software
  without prior written permission from Indiana University. For written
  permission, please contact Indiana University Advanced Research &
  Technology Institute.

* Products derived from this software may not be called "Caramel", nor may
  Indiana University, the University of Notre Dame or "Caramel" appear in
  their name, without prior written permission of Indiana University Advanced
  Research & Technology Institute.

Indiana University provides no reassurances that the source code provided does
not infringe the patent or any other intellectual property rights of any other
entity. Indiana University disclaims any liability to any recipient for claims
brought by any other entity based on infringement of intellectual property
rights or otherwise.

LICENSEE UNDERSTANDS THAT SOFTWARE IS PROVIDED "AS IS" FOR WHICH NO WARRANTIES
AS TO CAPABILITIES OR ACCURACY ARE MADE. INDIANA UNIVERSITY GIVES NO WARRANTIES
AND MAKES NO REPRESENTATION THAT SOFTWARE IS FREE OF INFRINGEMENT OF THIRD
PARTY PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS. INDIANA UNIVERSITY MAKES
NO WARRANTIES THAT SOFTWARE IS FREE FROM "BUGS", "VIRUSES", "TROJAN HORSES",
"TRAP DOORS", "WORMS", OR OTHER HARMFUL CODE. LICENSEE ASSUMES THE ENTIRE RISK
AS TO THE PERFORMANCE OF SOFTWARE AND/OR ASSOCIATED MATERIALS, AND TO THE
PERFORMANCE AND VALIDITY OF INFORMATION GENERATED USING SOFTWARE.
This package was debianized by Vladimir Prus <ghost@cs.msu.su> on
Copyright:

/+\  
+\Copyright 1993-2002 Christopher Seiwald and Perforce Software, Inc.  
/+\  

This is Release 2.4 of Jam/MR, a make-like program.

License is hereby granted to use this software and distribute it freely, as long as this copyright notice is retained and modifications are clearly marked.

ALL WARRANTIES ARE HEREBY DISCLAIMED.

Some portions are also:

Copyright 2001-2006 David Abrahams.
Copyright 2002-2006 Rene Rivera.
Copyright 2003-2006 Vladimir Prus.

Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)
<article Copyright Test
[quickbook 1.5]
copyright 1963-1965 Jane Doe, 2018 Joe Blow, John Coe]
copyright 1977,1985 Someone else]
]
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//
// (C) Copyright Ion Gaztanaga 2005-2012.
//
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE_1_0.txt or copy at
// http://www.boost.org/LICENSE_1_0.txt)
//
//
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
#ifndef BOOST_INTERPROCESS_DETAIL_MPL_HPP
#define BOOST_INTERPROCESS_DETAIL_MPL_HPP
#ifndef BOOST_CONFIG_HPP
# include <boost/config.hpp>
#endif
#
#define BOOST_INTERPROCESS_DETAIL_MPL_HPP
#endif

#ifndef BOOST_CONFIG_HPP
# include <boost/config.hpp>
#endif
# if defined(BOOST_HAS_PRAGMA_ONCE)
#pragma once

#include <cstdlib>

namespace boost {
    namespace interprocess {
        namespace ipcdetail {

            template <class T, T val>
            struct integral_constant {
                static const T value = val;
                typedef integral_constant<T, val> type;
            };

            template <bool C_>
            struct bool_ : integral_constant<bool, C_> {
                static const bool value = C_;}
            ;

            typedef bool_<true> true_;
            typedef bool_<false> false_;

            typedef true_ true_type;
            typedef false_ false_type;

            typedef char yes_type;
            struct no_type {
                char padding[8];
            };

            template <bool B, class T = void>
            struct enable_if_c {
                typedef T type;
            };

            template <class Cond, class T = void>
            struct enable_if : public enable_if_c<Cond::value, T> {
            };

            template <class Cond, class T = void>
            struct disable_if : public enable_if_c<!Cond::value, T> {
            };
        }
    }
}

Open Source Used In DNAC Platform Arica 3335
template <class T, class U>
class is_convertible
{
    typedef char true_t;
    class false_t { char dummy[2]; }; 
    static true_t dispatch(U);
    static false_t dispatch(...);
    static T trigger();

    public:
    static const bool value = sizeof(dispatch(trigger())) == sizeof(true_t);
};

template<
    bool C 
    , typename T1
    , typename T2
>
struct if_c
{
    typedef T1 type;
};

template<
    typename T1
    , typename T2
>
struct if_c<false,T1,T2>
{
    typedef T2 type;
};

template<
    typename T1
    , typename T2
    , typename T3
>
struct if_
{
    typedef typename if_c<false<0 != T1::value, T2, T3>::type type;
};

template <class Pair>
struct select1st
// : public std::unary_function<Pair, typename Pair::first_type>
{
    template<class OtherPair>
    const typename Pair::first_type& operator()(const OtherPair& x) const
Open Source Used In DNAC Platform Arica

```cpp
{ return x.first; }

const typename Pair::first_type& operator()(const typename Pair::first_type& x) const
{ return x; }
};

// identity is an extension: it is not part of the standard.
template <class T>
struct identity
//   : public std::unary_function<T,T>
{
  typedef T type;
  const T& operator()(const T& x) const
  { return x; }
};

template<std::size_t S>
struct ls_zeros
{
  static const std::size_t value = (S & std::size_t(1)) ? 0 : (1u + ls_zeros<(S >> 1u)>::value);
};

template<>
struct ls_zeros<0>
{
  static const std::size_t value = 0;
};

template<>
struct ls_zeros<1>
{
  static const std::size_t value = 0;
};

} //namespace ipcdetail
} //namespace interprocess
} //namespace boost

#endif   //#ifndef BOOST_INTERPROCESS_DETAIL_MPL_HPP

[Copyright 2007 John Maddock.
Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or copy at
http://www.boost.org/LICENSE_1_0.txt).
]

[section:mpl MPL Interoperability]
```
All the value based traits in this library conform to MPL's requirements for an Integral Constant type: that includes a number of rather intrusive workarounds for broken compilers.

Purely as an implementation detail, this means that `__true_type` inherits from `boost::mpl::true_`, `__false_type` inherits from `boost::mpl::false_`, and `__integral_constant<T, v>` inherits from `boost::mpl::integral_c<T,v>` (provided `T` is not `bool`)

[endsect]

Copyright 2006 - 2010 John Maddock and Paul A. Bristow.
Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt).

Boost.Geometry (aka GGL, Generic Geometry Library)

Copyright (c) 2007-2013 Barend Gehrels, Amsterdam, the Netherlands.
Copyright (c) 2008-2013 Bruno Lalande, Paris, France.
Copyright (c) 2009-2013 Mateusz Loskot, London, UK.

Use, modification and distribution is subject to the Boost Software License, Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt)

=============================================================================
Boost.Geometry (aka GGL, Generic Geometry Library)

Copyright (c) 2003 Dr John Maddock
Use, modification and distribution is subject to the
Boost Software License, Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt)

#include "licence_info.hpp"
#include "bcp_imp.hpp"
#include "fileview.hpp"
#include <fstream>
#include <iomanip>
#include <cstring>
#include <stdexcept>
#include <boost/lexical_cast.hpp>
#include <boost/filesystem/operations.hpp>
#include <boost/throw_exception.hpp>

// split_path is a small helper for outputting a path name,
// complete with a link to that path:

//
struct split_path
{
    const fs::path& root;
    const fs::path& file;
    split_path(const fs::path& r, const fs::path& f)
        : root(r), file(f){
private:
    split_path& operator=(const split_path&);
};

std::ostream& operator << (std::ostream& os, const split_path& p)
{
    os << "<a href=" << (p.root / p.file).string() << ">" << p.file.string() << "</a>";
    return os;
}

std::string make_link_target(const std::string& s)
{
    // convert an arbitrary string into something suitable
    // for an <a> name:
    std::string result;
    for(unsigned i = 0; i < s.size(); ++i)
    {
        result.append(1, static_cast<std::string::value_type>(std::isalnum(s[i]) ? s[i] : '_'));
    }
    return result;
}

void bcp_implementation::output_license_info()
{
    std::pair<const license_info*, int> licenses = get_licenses();

    std::map<int, license_data>::const_iterator i, j;
    i = m_license_data.begin();
    j = m_license_data.end();

    std::ofstream os(m_dest_path.string().c_str());
    if(!os)
    {
        std::string msg("Error opening ");
        msg += m_dest_path.string();
        return;
    }

    }
msg += " for output."
std::runtime_error e(msg);
boost::throw_exception(e);
}
os << "
""<!DOCTYPE HTML PUBLIC "/-//W3C//DTD HTML 4.0 Transitional//EN"">\n"
"<html>\n""<head>\n""<title>Boost Licence Dependency Information</title>
if(m_module_list.size() == 1)
{
    os << " for " << *(m_module_list.begin());
}
os << "</head>\n""<body>\n""<H1>Boost Licence Dependency Information</H1>
if(m_module_list.size() == 1)
{
    os << " for " << *(m_module_list.begin());
}
os << "</H1>\n""<H2>Contents</H2>
<pre><a href="#input">Input Information</a>
if(!m_bsl_summary_mode)
    os << "<a href="#summary">Licence Summary</a>
    os << "<a href="#details">Licence Details</a>
while(i != j)
{
    // title:
    os << " <A href="#" >> make_link_target(licenses.first[i->first].license_name)
        << "]" >> licenses.first[i->first].license_name >> "</a>\n";
    ++i;
}
os << "<a href="#files">Files with no recognised license</a>
    os << "<a href="#authors">Files with no recognised copyright holder</a>
if(!m_bsl_summary_mode)
{
    os << "Moving to the Boost Software License...\n" " <a href="#bsl-converted">Files that can be automatically converted to the Boost Software License</a>\n" " <a href="#to-bsl">Files that can be manually converted to the Boost Software License</a>\n" " <a href="#not-to-bsl">Files that can <b>NOT</b> be moved to the Boost Software License</a>\n" " <a href="#need-bsl-authors">Authors we need to move to the Boost Software License</a>\n"
"<a href="#copyright">Copyright Holder Information</a>
";
}

os <<
"<a href="#depend">File Dependency Information</a>
";
"</pre>";

//
// input Information:
//
if(m_scan_mode)
os << "<P>The following files were scanned for boost dependencies:<BR>
";
else
os << "<P>The following Boost modules were checked:<BR>
";

std::list<std::string>::const_iterator si = m_module_list.begin();
std::list<std::string>::const_iterator sj = m_module_list.end();
while(si != sj)
{
  os << *si << "<BR>
";
  ++si;
}
os << "</p><p>The Boost path was: <code>" << m_boost_path.string() << "</code></P>
";

//
// extract the boost version number from the boost directory tree,
// not from this app (which may have been built from a previous
// version):
//
fileview version_file(m_boost_path / "boost/version.hpp");
static const boost::regex version_regex(
  "^[[:blank:]]*#\[[[:blank:]]]*define\[[[:blank:]]+BOOST_VERSION\[[:blank:]]+\(\d+\)\]"
);
boost::cmatch what;
if(boost::regex_search(version_file.begin(), version_file.end(), what, version_regex))
{
  int version = boost::lexical_cast<int>(what.str(1));
os << "<p>The Boost version is: " << version / 100000 << "." << version / 100 % 100 << "." << version % 100
  "</p>\n";
}

//
// output each license:
//
for(i = m_license_data.begin();
i = m_license_data.end();
if(!m_bsl_summary_mode)
{
  //
  // start with the summary:

Licence Summary

<table>
<thead>
<tr>
<th>Licence Name</th>
<th>License Text</th>
<th>Authors</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example</td>
<td></td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Details:

- Authors: 5
- Files: 3

- Authors 1: John Doe
- Authors 2: Jane Smith
- Authors 3: David Lee

Files using this licence:

- File 1: Example1
- File 2: Example2
- File 3: Example3
files:

```
std::set<fs::path, path_less>::const_iterator m, n;
m = i->second.files.begin();
n = i->second.files.end();
while(m != n)
{
  os << split_path(m_boost_path, *m) << "\n";
  ++m;
}
```

```
else
{
  os << "This license is used by " << i->second.authors.size() << " authors (list omitted for brevity).\n";
  os << "This license applies to " << i->second.files.size() << " files (list omitted for brevity).\n";
}
```

```
++license_index;
++i;
```

```
// // Output list of files not found to be under license control:
//
// os << "Files With No Recognisable Licence\n"
// "The following " << m_unknown_licenses.size() << " files had no recognisable license information:\n";
std::set<fs::path, path_less>::const_iterator i2, j2;
i2 = m_unknown_licenses.begin();
j2 = m_unknown_licenses.end();
while(i2 != j2)
{
  os << split_path(m_boost_path, *i2) << "\n";
  ++i2;
}
```

```
os << "Files With No Recognisable Copyright Holder\n"
"The following " << m_unknown_authors.size() << " files had no recognisable copyright holder:\n";
i2 = m_unknown_authors.begin();
j2 = m_unknown_authors.end();
while(i2 != j2)
{
  os << split_path(m_boost_path, *i2) << "\n";
  ++i2;
}
```

```
os << "\n";
```
if(!m_bsl_summary_mode)
{
  //
  // Output list of files that have been moved over to the Boost Software License, along with enough information for human verification.
  //
  os << "<h2><a name="bsl-converted"></a>Files that can be automatically converted to the Boost Software License</h2><br/>
";
  << "<p>The following " << m_converted_to_bsl.size() << " files can be automatically converted to the Boost Software License, but require manual verification before they can be committed to CVS:<br/>
";
  if (!m_converted_to_bsl.empty())
  {
    typedef std::map<fs::path, std::pair<std::string, std::string>, path_less<br />
    ::const_iterator conv_iterator;
    conv_iterator i = m_converted_to_bsl.begin(),
    ie = m_converted_to_bsl.end();
    int file_num = 1;
    while (i != ie)
    {
      os << "<p><br [" << file_num << "] File: <tt>" << split_path(m_boost_path, i->first)
      " <br><pre>"
      "<i>" << i->second.first << "<pre></td><i><td><pre>
      "<i>" << i->second.second << "<pre></td><i></tr></table><i>"
      
      ++i;
      ++file_num;
    }
  }
  //
  // Output list of files that could be moved over to the Boost Software License
  //
  os << "<h2><a name="to-bsl"></a>Files that could be converted to the Boost Software License</h2><br/>
";
  "<p>The following " << m_can_migrate_to_bsl.size() << " files could be manually converted to the Boost Software License, but have not yet been:<br/>
";
  i2 = m_can_migrate_to_bsl.begin();
  j2 = m_can_migrate_to_bsl.end();
  while(i2 != j2)
  {
    os << split_path(m_boost_path, *i2) << "<br><i>"
    
    ++i2;
  }
  os << "</p></BLOCKQUOTE>";
  //
  // Output list of files that cannot be moved over to the Boost Software License
  //
  os << "<h2><a name="not-to-bsl"></a>Files that can NOT be converted to the Boost Software License</h2><br/>
";
  "<p>The following " << m_cannot_migrate_to_bsl.size() << " files cannot be converted to the Boost Software License:<br/>
";
License because we need the permission of more authors:

```c++
// Output list of authors that we need permission for to move to the BSL
//
std::copy(m_authors_for_bsl_migration.begin(), m_authors_for_bsl_migration.end(),
          std::ostream_iterator<std::string>(os, "\n"));
```

"Permission of the following authors is needed before we can convert to the Boost Software License. The list of authors that have given their permission is contained in more/blanket-permission.txt"

```c++
// output a table of copyright information:
//
std::map<std::string, std::set<fs::path, path_less>> m_author_data;

std::map<fs::path, fs::path, path_less>::const_iterator ad, ead;
ad = m_author_data.begin();
ead = m_author_data.end();
while(ad != ead)
{
    os << "<tr><td>" << ad->first << "</td><td>
    std::set<fs::path, path_less>::const_iterator fi, efi;
    fi = ad->second.begin();
    efi = ad->second.end();
    while(fi != efi)
    {
        os << split_path(m_boost_path, *fi) << " \n";
        ++fi;
    }
    os << "</td></tr>
    ++ad;
}
```

"File Dependency Information"

```c++
```

// output file dependency information:
//
std::map<fs::path, fs::path, path_less>::const_iterator dep, last_dep;
std::set<fs::path, path_less>::const_iterator fi, efi;
fi = m_copy_paths.begin();
efi = m_copy_paths.end();
// if in summary mode, just figure out the "bad" files and print those only:
std::set<fs::path, path_less> bad_paths;
if(m_bsl_summary_mode)
{
    bad_paths.insert(m_unknown_licenses.begin(), m_unknown_licenses.end());
    bad_paths.insert(m_unknownAuthors.begin(), m_unknownAuthors.end());
    bad_paths.insert(m_can_migrate_to_bsl.begin(), m_can_migrate_to_bsl.end());
    bad_paths.insert(m_cannot_migrate_to_bsl.begin(), m_cannot_migrate_to_bsl.end());
    typedef std::map<fs::path, std::pair<std::string, std::string>, path_less>
      ::const_iterator conv_iterator;
    conv_iterator i = m_converted_to_bsl.begin(),
       ie = m_converted_to_bsl.end();
    while(i != ie)
    {
        bad_paths.insert(i->first);
        ++i;
    }
    fi = bad_paths.begin();
    efi = bad_paths.end();
    os << "<P>For brevity, only files not under the BSL are shown</P><br>
        
    }
while(fi != efi)
{
    os << split_path(m_boost_path, *fi);
    dep = m_dependencies.find(*fi);
    last_dep = m_dependencies.end();
    std::set<fs::path, path_less> seen_deps;
    if (dep != last_dep)
        while(true)
        {
            os << " -> ";
            if(fs::exists(m_boost_path / dep->second))
                os << split_path(m_boost_path, dep->second);
            else if(fs::exists(dep->second))
                os << split_path(fs::path(), dep->second);
            else
                os << dep->second.string();
            if(seen_deps.find(dep->second) != seen_deps.end())
                { os << " <I>(Circular dependency!)</I>";
                    break; // circular dependency!!!
                }
            seen_deps.insert(dep->second);
            last_dep = dep;
            dep = m_dependencies.find(dep->second);
if((dep == m_dependencies.end()) || (0 == compare_paths(dep->second, last_dep->second)))
    break;
}
    os << "\n"
    ++fi;
}
    os << "</pre></BLOCKQUOTE>\n"

    os << "</body></html>\n"

    if(!os)
        {
            std::string msg("Error writing to ");
            msg += m_dest_path.string();
            msg += ":
            std::runtime_error e(msg);
            boost::throw_exception(e);
        }
    }

1.987 depd 1.1.2 1.1.2
1.987.1 Available under license :
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.988 netty 4.0.48.Final
1.988.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must
be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, 5th Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuate of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.
Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2011 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:
* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/protobuf

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/snappy

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-
benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/caliper

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
Open Source Used In DNAC Platform Arica 3363

modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

(BSD License: http://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Webbit nor the names of
its contributors may be used to endorse or promote products
derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT,
INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA,
OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGligence OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
/
* Copyright (c) 2004-2007 QOS.ch
* All rights reserved.
* 
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
  * "Software"), to deal in the Software without restriction, including
  * without limitation the rights to use, copy, modify, merge, publish,
  * distribute, sublicense, and/or sell copies of the Software, and to
  * permit persons to whom the Software is furnished to do so, subject to
  * the following conditions:
  *
  * The above copyright notice and this permission notice shall be
  * included in all copies or substantial portions of the Software.
  *
  * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
  * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
  * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
  * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
  * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
  * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
  * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.990 fast-levenshtein 2.0.6

1.990.1 Available under license:

(MIT License)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.991 fbjs 0.8.17 0.8.17

1.991.1 Available under license :
MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.992 is-date-object 1.0.2

1.992.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.993 hk2-utils 2.6.1
1.993.1 Available under license:
 # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.
"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:
a) it must be made available under this Agreement, or if the
Program (i) is combined with other material in a separate file or
files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in
Exhibit A of this Agreement, then the Program may be made available
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product
offering should do so in a manner which does not create potential
liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program
in a commercial product offering. The obligations in this section do not
apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified
Contributor must: a) promptly notify the Commercial Contributor in
writing of such claim, and b) allow the Commercial Contributor to control,
and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may
participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial
product offering, Product X. That Contributor is then a Commercial
Contributor. If that Commercial Contributor then makes performance
claims, or offers warranties related to Product X, those performance
claims and warranties are such Commercial Contributor's responsibility
alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance
claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of
time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium.
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version"; you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.
One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.
## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0
## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.994 memoize-one 4.1.0 4.1.0

#### 1.994.1 Available under license :

MIT License

Copyright (c) 2017 Alexander Reardon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.995 object-map 1.0.1 1.0.1
1.995.1 Available under license:

Copyright (c) 2014-2017, Jon Schlinkert, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.996 hadoop-annotations 3.2.2
1.996.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:

* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com
The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4, which has the following notices:

The Netty Project
=================

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jboscache/experimental/jsr166/
This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java
This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBOSS Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
  Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
* TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
* ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by
"GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of
Xerces2 Java Parser 2.9.1,
which has the following notices:
* -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:

* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors
Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 2008-2015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 2007-2008 Robert Niederreiter, Jens Klein
# Copyright 2004-2005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64,
LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * - Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor
the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/*++
 *
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 *
 * Abstract: The main routine
 *
/--=*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/[lz4.h,lz4.c,lz4hc.h,lz4hc.c],
/*
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

 * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at :
- LZ4 source repository : http://code.google.com/p/lz4/
- LZ4 public forum : https://groups.google.com/forum/#!forum/lz4c

*/

For hadoop-common-project/hadoop-common/src/main/native/gtest
---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

----------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

----------------------------------------------------------------------

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
```
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js
```

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:
```
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
datatables v1.10.8
```
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
--------------------------------------------------------------------------------------------------------------------------
Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

--------------------------------------------------------------------------------

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7
MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

For:
oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as
documented below:

====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js
--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js
--------------------------------------------------------------------------------
D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQldb Database 2.3.4

(HSQL License)
"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0
1. Definitions.

1.1.Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2.Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities,
You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that
Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the

Open Source Used In DNAC Platform Arica 3427
recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license.
4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to
termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control
laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions
thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification;
or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under
intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software
exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or
new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original
Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3 If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4 In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of,
and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

--------------------------------------------------------------------------------
(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs
of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions)
under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0

dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a
Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

- Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.
- Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats.

All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use
of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

MPL 2.0

Mozilla Public License

Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or
earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.
2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of
2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered
Software. If the Larger Work is a combination of Covered Software with a work
governed by one or more Secondary Licenses, and the Covered Software is not
Incompatible With Secondary Licenses, this License permits You to additionally
distribute such Covered Software under the terms of such Secondary License(s),
so that the recipient of the Larger Work may, at their option, further
distribute the Covered Software under the terms of either this License or such
Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including
copyright notices, patent notices, disclaimers of warranty, or limitations of
liability) contained within the Source Code Form of the Covered Software, except
that You may alter any license notices to the extent required to remedy known
factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity
or liability obligations to one or more recipients of Covered Software. However,
You may do so only on Your own behalf, and not on behalf of any Contributor. You
must make it absolutely clear that any such warranty, support, indemnity, or
liability obligation is offered by You alone, and You hereby agree to indemnify
every Contributor for any liability incurred by such Contributor as a result of
warranty, support, indemnity or liability terms You offer. You may include
additional disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with
respect to some or all of the Covered Software due to statute, judicial order,
or regulation then You must: (a) comply with the terms of this License to the
maximum extent possible; and (b) describe the limitations and the code they
affect. Such description must be placed in a text file included with all
distributions of the Covered Software under this License. Except to the extent
prohibited by statute or regulation, such description must be sufficiently
detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You
fail to comply with any of its terms. However, if You become compliant, then the
rights granted under this License from a particular Contributor are reinstated
(a) provisionally, unless and until such Contributor explicitly and finally
terminates Your grants, and (b) on an ongoing basis, if such Contributor fails
to notify You of the non-compliance by some reasonable means prior to 60 days
after You have come back into compliance. Moreover, Your grants from a
particular Contributor are reinstated on an ongoing basis if such Contributor 
notifies You of the non-compliance by some reasonable means, this is the first 
time You have received notice of non-compliance with this License from such 
Contributor, and You become compliant prior to 30 days after Your receipt of the 
notice.

5.2. If You initiate litigation against any entity by asserting a patent 
infringement claim (excluding declaratory judgment actions, counter-claims, and 
cross-claims) alleging that a Contributor Version directly or indirectly 
infringes any patent, then the rights granted to You by any and all Contributors 
for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user 
license agreements (excluding distributors and resellers) which have been 
validly granted by You or Your distributors under this License prior to 
termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without 
warranty of any kind, either expressed, implied, or statutory, including, 
without limitation, warranties that the Covered Software is free of defects, 
merchantable, fit for a particular purpose or non-infringing. The entire risk as 
to the quality and performance of the Covered Software is with You. Should any 
Covered Software prove defective in any respect, You (not any Contributor) 
assume the cost of any necessary servicing, repair, or correction. This 
disclaimer of warranty constitutes an essential part of this License. No use of 
any Covered Software is authorized under this License except under this 
 disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including 
negligence), contract, or otherwise, shall any Contributor, or anyone who 
distributes Covered Software as permitted above, be liable to You for any 
direct, indirect, special, incidental, or consequential damages of any character 
including, without limitation, damages for lost profits, loss of goodwill, work 
stoppage, computer failure or malfunction, or any and all other commercial 
damages or losses, even if such party shall have been informed of the 
possibility of such damages. This limitation of liability shall not apply to 
liability for death or personal injury resulting from such partys negligence 
to the extent applicable law prohibits such limitation. Some jurisdictions do 
not allow the exclusion or limitation of incidental or consequential damages, so 
this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a
jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.
If it is not possible or desirable to put the notice in a particular file, then
You may include the notice in a location (such as a LICENSE file in a relevant
directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined
by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the
following license:
JDOM 1.1
--------------------------------------------------------------------------------
/*--
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions, and the disclaimer that follows
these conditions in the documentation and/or other materials
provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products
derived from this software without prior written permission. For
written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor
may "JDOM" appear in their name, without prior written permission
from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the
software itself an acknowledgement equivalent to the following:
"This product includes software developed by the
JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos
THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter AT jdom DOT org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

--------------------------------------------------------------------------------

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE
TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or
form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the
open source used in dac platform arica 3459

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological
measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.
b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor
agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous
a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.
Except for the limited purpose of indicating to the public that the
Work is licensed under the CCPL, Creative Commons does not authorize
the use by either party of the trademark "Creative Commons" or any
related trademark or logo of Creative Commons without the prior
written consent of Creative Commons. Any permitted use will be in
compliance with Creative Commons' then-current trademark usage
guidelines, as may be published on its website or otherwise made
available upon request from time to time. For the avoidance of doubt,
this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

--------------------------------------------------------------------------------

For: hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
/server/datanode/checker/AbstractFuture.java and
hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
/server/datanode/checker/TimeoutFuture.java

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not
use this file except in compliance with the License. You may obtain a copy of
the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations under
the License.

1.997 codemirror 5.58.2 5.58.2
1.997.1 Available under license :
MIT License

Copyright (C) 2017 by Marijn Haverbeke <marijnh@gmail.com> and others

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.998 blob 0.0.5 0.0.5
1.998.1 Available under license :
MIT License

Copyright (C) 2014 Rase-

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.999 acorn-jsx 3.0.1
1.999.1 Available under license :
Copyright (C) 2012-2014 by Ingvar Stepanyan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1000 send 0.13.2 0.13.2
1.1000.1 Available under license :
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1001 uuid 3.3.3 3.3.3
1.1001.1 Available under license :
The MIT License (MIT)

Copyright (c) 2010-2016 Robert Kieffer and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1002 babel-helper-get-function-arity 6.24.1

1.1002.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywritens noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE USE OR OTHER DEALINGS IN THE SOFTWARE.
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1003 array-unique 0.2.1
1.1003.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1004 stax 3.1.4

1.1004.1 Available under license:

No license file was found, but licenses were detected in source scan.

/* Stax2 API extension for Streaming API for XML processing (StAX).
 *
 * Copyright (c) 2006- Tatu Saloranta, tatu.salaranta@iki.fi
 *
 * Licensed under the License specified in the file LICENSE which is
 * included with the source code.
 * You may not use this file except in compliance with the License.
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-jar/org/codehaus/stax2/ri/dom/DOMWrappingReader.java

No license file was found, but licenses were detected in source scan.

/* Stax2 API extension for Streaming Api for Xml processing (StAX).
 *
 * Copyright (c) 2006- Tatu Saloranta, tatu.salaranta@iki.fi
 *
 * Licensed under the License specified in the file LICENSE which is
 * included with the source code.
 * You may not use this file except in compliance with the License.
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-jar/org/codehaus/stax2/ri/DOMWrappingWriter.java
No license file was found, but licenses were detected in source scan.

/* Stax2 API extension for Streaming Api for Xml processing (StAX).
 * Copyright (c) 2006- Tatu Saloranta, tatu.saloranta@iki.fi
 * Licensed under the License specified in file LICENSE, included with
 * the source code.
 * You may not use this file except in compliance with the License.
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/Stax2WriterAdapter.java
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/Stax2WriterImpl.java
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/Stax2ReaderImpl.java
No license file was found, but licenses were detected in source scan.

/* StAX2 extension for StAX API (JSR-173).
 * Copyright (c) 2005- Tatu Saloranta, tatu.saloranta@iki.fi
 * Licensed under the License specified in file LICENSE, included with
 * the source code.
 * You may not use this file except in compliance with the License.
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/typed/Base64DecoderBase.java
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/typed/CharArrayBase64Decoder.java
* /opt/cola/permits/1129090477_1612176217.67/0/stax2-api-3-1-4-sources-
  jar/org/codehaus/stax2/ri/typed/StringBase64Decoder.java
No license file was found, but licenses were detected in source scan.

/* Reference Implementation of
 * Stax2 extension API (for basic Stax API, JSR-173)
 *
 * Copyright (c) 2005- Tatu Saloranta, tatu.saloranta@iki.fi
 *
 * Licensed under the License specified in file LICENSE, included with
 * the source code.
 * You may not use this file except in compliance with the License.
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/* Stax2 extension for basic Stax API (JSR-173).
 *
 * Copyright (c) 2005- Tatu Saloranta, tatu.saloranta@iki.fi
 *
 * Licensed under the License specified in file LICENSE, included with
 * the source code.
 * You may not use this file except in compliance with the License.
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
jar/org/codehaus/stax2/XMLStreamReader2.java
/* Reference Implementation of
* Stax2 extension API (for basic Stax API, JSR-173)
* "Reference Implementation of Stax2 extension API (for basic Stax API, JSR-173)"
* Copyright (c) 2008- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in file LICENSE, included with
* the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/* Reference Implementation of
* Stax2 extension API (for basic Stax API, JSR-173)
* "Reference Implementation of Stax2 extension API (for basic Stax API, JSR-173)"
* Copyright (c) 2008- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in file LICENSE, included with
* the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/* Woodstox XML processor
* "Woodstox XML processor"
* Copyright (c) 2004- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in the file LICENSE which is
* included with the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/* Woodstox XML processor
* "Woodstox XML processor"
* Copyright (c) 2004- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in the file LICENSE which is
* included with the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

Found in path(s):
/* Reference Implementation of
* Stax2 extension API (for basic Stax API, JSR-173)
* "Reference Implementation of Stax2 extension API (for basic Stax API, JSR-173)"
* Copyright (c) 2008- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in file LICENSE, included with
* the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/* Woodstox XML processor
* "Woodstox XML processor"
* Copyright (c) 2004- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in the file LICENSE which is
* included with the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/* Reference Implementation of
* Stax2 extension API (for basic Stax API, JSR-173)
* "Reference Implementation of Stax2 extension API (for basic Stax API, JSR-173)"
* Copyright (c) 2008- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in file LICENSE, included with
* the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/* Reference Implementation of
* Stax2 extension API (for basic Stax API, JSR-173)
* "Reference Implementation of Stax2 extension API (for basic Stax API, JSR-173)"
* Copyright (c) 2008- Tatu Saloranta, tatu.saloranta@iki.fi
* Licensed under the License specified in file LICENSE, included with
* the source code.
* You may not use this file except in compliance with the License.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
1.1005 is-my-ip-valid 1.0.0 1.0.0

1.1005.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1006 json-schema-faker 0.5.0-rcv.34 0.5.0-rcv.34

1.1006.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2018 Alvaro Cabrera

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1007 aws-java-sdk-sts 1.11.837

1.1007.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/internal/STSProfileCredentialsService.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2020 Amazon Technologies, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/auth/SessionCredentialsProviderFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/ShouldDoBlockingSessionRefresh.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/RefreshableTask.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/ShouldDoAsyncSessionRefresh.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/DaemonThreadFactory.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/SessionCredentialsHolder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */
/**
 * <p>
 * An IAM policy in JSON format that you want to use as an inline session policy.
You must pass an inline or managed session policy to this operation. You can pass a single JSON policy document to use as an inline session policy. You can also specify up to 10 managed policies to use as managed session policies.

This parameter is optional. However, if you do not pass any session policies, then the resulting federated user session has no permissions.

When you pass session policies, the session permissions are the intersection of the IAM user policies and the session policies that you pass. This gives you a way to further restrict the permissions for a federated user.

You cannot use session policies to grant more permissions than those that are defined in the permissions policy of the IAM user. For more information, see Session Policies in the IAM User Guide.

The resulting credentials can be used to access a resource that has a resource-based policy. If that policy specifically references the federated user session in the Principal element of the policy, the session has the permissions allowed by the policy. These permissions are granted in addition to the permissions that are granted by the session policies.

The plain text that you use for both inline and managed session policies can't exceed 2,048 characters. The JSON policy characters can be any ASCII character from the space character to the end of the valid character list (through \u00FF). It can also include the tab ( ), linefeed ( ), and carriage return ( ) characters.

An AWS conversion compresses the passed session policies and session tags into a packed binary format that has a separate limit. Your request can fail for this limit even if your plain text meets the other requirements. The PackedPolicySize response element indicates by percentage how close the policies and tags for your request are to the upper size limit.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 permissions
 * and limitations under the License.
 */
/**
 *
 * Returns a set of temporary security credentials (consisting of an access key ID, a secret access key, and a
 * security token) for a federated user. A typical use is in a proxy application that gets temporary security
 * credentials on behalf of distributed applications inside a corporate network. You must call the
 * <code>GetFederationToken</code> operation using the long-term security credentials of an IAM user. As a
 * result,
 * this call is appropriate in contexts where those credentials can be safely stored, usually in a server-based
 * application. For a comparison of <code>GetFederationToken</code> with the other API operations that
 produce
 * temporary credentials, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html">Requesting
 Temporary
 * Security Credentials</a> and <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html#stsapi_comparison"
 >Comparing the AWS STS API operations</a> in the <i>IAM User Guide</i>.
 *
 * <p>
 * <note>
 * You can create a mobile-based or browser-based app that can authenticate users using a web identity provider
 like
 * Login with Amazon, Facebook, Google, or an OpenID Connect-compatible identity provider. In this case, we
 * recommend that you use <a href="http://aws.amazon.com/cognito/">Amazon Cognito</a> or
 * <code>AssumeRoleWithWebIdentity</code>. For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html#api_assumerolewithwebidentity"
 >Federation Through a Web-based Identity Provider</a> in the <i>IAM User Guide</i>.
 * </note>
 */
You can also call `<code>GetFederationToken</code>` using the security credentials of an AWS account root user, but we do not recommend it. Instead, we recommend that you create an IAM user for the purpose of the proxy application. Then attach a policy to the IAM user that limits federated users to only the actions and resources that they need to access. For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html">IAM Best Practices</a> in the IAM User Guide.

**Session duration**

The temporary credentials are valid for the specified duration, from 900 seconds (15 minutes) up to a maximum of 129,600 seconds (36 hours). The default session duration is 43,200 seconds (12 hours). Temporary credentials that are obtained by using AWS account root user credentials have a maximum duration of 3,600 seconds (1 hour).

**Permissions**

You can use the temporary credentials created by `<code>GetFederationToken</code>` in any AWS service except the following:

- You cannot call any IAM operations using the AWS CLI or the AWS API.
- You cannot call any STS operations except `<code>GetCallerIdentity</code>`.

You must pass an inline or managed session policy to this operation. You can pass a single JSON policy document to use as an inline session policy. You can also specify up to 10 managed policies to use as managed session policies. The plain text that you use for both inline and managed session policies can't exceed 2,048 characters.
Though the session policy parameters are optional, if you do not pass a policy, then the resulting federated user session has no permissions. When you pass session policies, the session permissions are the intersection of the IAM user policies and the session policies that you pass. This gives you a way to further restrict the permissions for a federated user. You cannot use session policies to grant more permissions than those that are defined in the permissions policy of the IAM user. For more information, see Session Policies in the IAM User Guide. For information about using GetFederationToken to create temporary security credentials, see GetFederationToken—Federation Through a Custom Identity Broker.

You can use the credentials to access a resource that has a resource-based policy. If that policy specifically references the federated user session in the Principal element of the policy, the session has the permissions allowed by the policy. These permissions are granted in addition to the permissions granted by the session policies.

(Optional) You can pass tag key-value pairs to your session. These are called session tags. For more information about session tags, see Passing Session Tags in STS in the IAM User Guide.

An administrator must grant you the permissions necessary to pass session tags. The administrator can also create granular permissions to allow you to pass only specific session tags. For more information, see Tutorial: Using Tags for Attribute-Based Access Control in the IAM User Guide.

Tag key–value pairs are not case sensitive, but case is preserved. This means that you cannot have separate <code>Department</code> and <code>department</code> tag keys. Assume that the user that you are federating has the session tag <code>Department</code>=<code>Marketing</code> and you pass the session tag <code>Department</code>=<code>engineering</code> in the request. The session tag passed in the request takes precedence over the user tag.
* @param getFederationTokenRequest
* @return A Java Future containing the result of the GetFederationToken operation returned by the service.
* @sample AWSSecurityTokenServiceAsync.GetFederationToken
* @see <a href="http://docs.aws.amazon.com/goto/WebAPI/sts-2011-06-15/GetFederationToken" target="_top">AWS API Documentation</a>

*/

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/AWSSecurityTokenServiceAsync.java

No license file was found, but licenses were detected in source scan.

"*
* Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
* the License. A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
* and limitations under the License.
*/
/**
 * Returns a set of temporary security credentials (consisting of an access key ID, a secret access key, and a
 * security token) for a federated user. A typical use is in a proxy application that gets temporary security
 * credentials on behalf of distributed applications inside a corporate network. You must call the
 * <code>GetFederationToken</code> operation using the long-term security credentials of an IAM user. As a result,
 * this call is appropriate in contexts where those credentials can be safely stored, usually in a server-based
 * application. For a comparison of <code>GetFederationToken</code> with the other API operations that produce
 */

<note>
</note>

<p>
</p>
* You can create a mobile-based or browser-based app that can authenticate users using a web identity provider like
  * Login with Amazon, Facebook, Google, or an OpenID Connect-compatible identity provider. In this case, we recommend that you use <a href="http://aws.amazon.com/cognito/">Amazon Cognito</a> or <code>AssumeRoleWithWebIdentity</code>. For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html#api_assumerolewithwebidentity">Federation Through a Web-based Identity Provider</a> in the <i>IAM User Guide</i>.
  *
  * You can also call <code>GetFederationToken</code> using the security credentials of an AWS account root user, but we do not recommend it. Instead, we recommend that you create an IAM user for the purpose of the proxy application. Then attach a policy to the IAM user that limits federated users to only the actions and resources that they need to access. For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html">IAM Best Practices</a> in the <i>IAM User Guide</i>.

* **Session duration**

  * The temporary credentials are valid for the specified duration, from 900 seconds (15 minutes) up to a maximum of 129,600 seconds (36 hours). The default session duration is 43,200 seconds (12 hours). Temporary credentials that are obtained by using AWS account root user credentials have a maximum duration of 3,600 seconds (1 hour).

* **Permissions**

  * You can use the temporary credentials created by <code>GetFederationToken</code> in any AWS service except the following:
    * You cannot call any IAM operations using the AWS CLI or the AWS API.
    * You cannot call any STS operations except <code>GetCallerIdentity</code>. 

You must pass an inline or managed session policy to this operation. You can pass a single JSON policy document to use as an inline session policy. You can also specify up to 10 managed policies to use as managed session policies. The plain text that you use for both inline and managed session policies can't exceed 2,048 characters.

Though the session policy parameters are optional, if you do not pass a policy, then the resulting federated user session has no permissions. When you pass session policies, the session permissions are the intersection of the IAM user policies and the session policies that you pass. This gives you a way to further restrict the permissions for a federated user. You cannot use session policies to grant more permissions than those that are defined in the permissions policy of the IAM user. For more information, see Session Policies in the *IAM User Guide*. For information about using *GetFederationToken* to create temporary security credentials, see [GetFederationToken—Federation Through a Custom Identity Broker](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html#api_getfederationToken)

You can use the credentials to access a resource that has a resource-based policy. If that policy specifically references the federated user session in the *Principal* element of the policy, the session has the permissions allowed by the policy. These permissions are granted in addition to the permissions granted by the session policies.

* Tags
* (Optional) You can pass tag key-value pairs to your session. These are called session tags. For more information about session tags, see [Passing Session Tags in STS](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_session-tags.html)

An administrator must grant you the permissions necessary to pass session tags. The administrator can also create granular permissions to allow you to pass only specific session tags. For more information, see [Tutorial: Using Tags for Attribute-Based Access Control](https://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial_attribute-based-access-control.html).
**Tag key–value pairs are not case sensitive, but case is preserved. This means that you cannot have separate**
*Department* and *department* tag keys. Assume that the user that you are federating has
*the Department* tag and you pass the *department* session tag. *Department* and *department* are not saved as
*separate tags, and the session tag passed in the request takes precedence over the user tag.*

@throws MalformedPolicyDocumentException
   The request was rejected because the policy document was malformed. The error message describes the
   specific error.

@throws PackedPolicyTooLargeException
   The request was rejected because the total packed size of the session policies and session tags combined
   was too large. An AWS conversion compresses the session policy document, session policy ARNs, and
   session tags into a packed binary format that has a separate limit. The error message indicates by percentage how
   close the policies and tags are to the upper size limit. For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_session-tags.html">Passing Session Tags in STS</a> in the *IAM User Guide*.</p>

You could receive this error even though you meet other defined session policy and session tag limits.
For more information, see <a href="https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_enable-regions.html">IAM and STS Entity Character Limits</a> in the *IAM User Guide*.

@throws RegionDisabledException
   STS is not activated in the requested region for the account that is being asked to generate credentials.
   The account administrator must use the IAM console to activate STS in that region. For more information,

@sample AWSSecurityTokenService.GetFederationToken

@see <a href="http://docs.aws.amazon.com/goto/WebAPI/sts-2011-06-15/GetFederationToken">AWS API Documentation</a>

/ opt/ cola/ permits/ 11377 60099_ 1614121467.9/ 0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/AWSSecurityTokenServiceClient.java

Found in path(s):
/api/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/AWSSecurityTokenService.java

/api/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/AWSSecurityTokenServiceClient.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2020 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/auth/policy/STSActions.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithWebIdentityRequest.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/ExpiredTokenException.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithSAMLResult.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithWebIdentityResult.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithSAMLResult.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithWebIdentityResult.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-jar/com/amazonaws/services/securitytoken/model/AssumeRoleWithSAMLResult.java
/*
 * Copyright 2011-2020 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * 
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/NoSessionSupportCredentials.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/STSSessionCredentialsProvider.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/WebIdentityFederationSessionCredentialsProvider.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/STSSessionCredentials.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/STSSessionCredentialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/auth/STSSessionCredentialsWithWebIdentitySessionCredentialsProvider.java
* /opt/cola/permits/1137760099_1614121467.9/0/aws-java-sdk-sts-1-11-837-sources-
jar/com/amazonaws/services/securitytoken/RegionalEndpointsOptionResolver.java

1.1008 oas-linter 3.2.1 3.2.1

1.1008.1 Available under license:

BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1009 y18n 4.0.3 4.0.3
1.1009.1 Available under license :
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

1.1010 window-size 0.1.4 0.1.4
1.1010.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.


1.1011.1 Available under license :
(The MIT License)

Copyright (C) 2011-2015 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1012 babel-plugin-transform-es2015-object-super 6.24.1
1.1012.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1013 @babel/helper-module-transforms

7.16.0

1.1013.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1014 akka 2.5.31

1.1014.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>  

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, 5th Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.
The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from
liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc. (http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2011 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified version of 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD Style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product contains a modified version of 'Webbit', a Java event based WebSocket and HTTP server:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit
This product optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'JBoss Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-logging.txt (GNU LGPL 2.1)
* HOMEPAGE:
This product optionally depends on 'Apache Felix', an open source OSGi framework implementation, which can be obtained at:

* http://anonsvn.jboss.org/repos/common/common-logging-spi/

* LICENSE:
  *
  * license/LICENSE.felix.txt (Apache License 2.0)
  *
* HOMEPAGE:
  *
  * http://felix.apache.org/

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  *
  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  *
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  *
  * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(BSD License: http://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Webbit nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
/
* Copyright (c) 2004-2007 QOS.ch
* All rights reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted toLicensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1015 netty-transport-native-epoll

4.1.59.Final

1.1015.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
/*

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollChannelConfig.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollMode.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollRecvByteAllocatorStreamingHandle.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-
jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Set the [ @code TCP_MD5SIG ] option on the socket. See [ @code linux/tcp.h ] for more details.
* Keys can only be set on, not read to prevent a potential leak, as they are confidential.
* Allowing them being read would mean anyone with access to the channel could get them.
*/
/**
 * Set the \code TCP_QUICKACK\ option on the socket.
 * See \html \a href="https://linux.die.net/man/7/tcp">TCP_QUICKACK</a>\ for more details.
 */

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*/
Open Source Used In DNAC Platform Arica  3523

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/netty_epoll_linuxsocket.h
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/netty_epoll_linuxsocket.c
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/NativeStaticallyReferencedJniMethods.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/LinuxSocket.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
 * Set the { @code TCP_MD5SIG} option on the socket. See { @code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2013 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/Native.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/netty_epoll_native.c

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollChannelOption.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollServerSocketChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollDatagramChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/NativeDatagramPacketArray.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollTcpInfo.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/AbstractEpollChannel.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollEventLoopGroup.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources-jar/io/netty/channel/epoll/EpollEventLoopGroup.java

Open Source Used In DNAC Platform Arica 3524
jar/io/netty/channel/epoll/EpollEventLoop.java
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources.jar/io/netty/channel/epoll/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1136534488_1613780692.52/0/netty-transport-native-epoll-4-1-59-final-sources.jar/io/netty/channel/epoll/EpollDatagramChannelConfig.java

1.1016 drill-storage-http 1.19.0
1.1016.1 Available under license:

Drill : Contrib : Storage : HTTP
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Apache Avro Mapred API
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses...
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1018 hbase 2.4.2
1.1018.1 Available under license :

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

This project bundles a copy of the JQuery minified javascript library version 1.8.3 under the terms of the MIT license.

Copyright 2012 jQuery Foundation and other contributors
http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
Software”), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

----

This project bundles portions of the 'JQuery' project under the terms of the MIT license.

Copyright 2012 jQuery Foundation and other contributors
http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
This project bundles a derivative of portions of the 'Asciidoctor' project under the terms of the MIT license.

The MIT License
Copyright (C) 2012-2015 Dan Allen, Ryan Waldron and the Asciidoctor Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This project incorporates portions of the 'Protocol Buffers' project available under a '3-clause BSD' license.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

----

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions
a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography;
a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

   a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
   b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a
modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You
must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author’s honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author’s honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the...
applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in
compliance with Creative Commons’ then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

Apache HBase
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14 and v21, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/util/Bytes.java'
Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

--
This product includes portions of Jetty project, specially
'hbase-shaded-hbase-shaded-testing-
util/src/main/resources/org/apache/hadoop/hbase/shaded/org/mortbay/jetty/webapp/webdefault.xml'
In Ruby's source distribution, this would describe a number of C source files
that have different licenses than Ruby itself. None of those apply to JRuby,
so we have this file here as a placeholder.

For details of licensing of this collective work, see LICENSE

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

${project.name}
Copyright ${project.inceptionYear}-${build.year} The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1019 picomatch 2.2.2
1.1019.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1020 hpkp 1.2.0 1.2.0
1.1020.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2016 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1021 invert-kv 2.0.0 2.0.0
1.1021.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1022 javax-servlet-jsp-api 2.1
1.1022.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof),
with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third
party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version
of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding
declaratory judgment actions) against Initial Developer or a
Contributor (the Initial Developer or Contributor against whom You
assert such claim is referred to as "Participant") alleging that
the Participant Software (meaning the Contributor Version where
the Participant is a Contributor or the Original Software where
the Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or
indirectly to You by such Participant, the Initial Developer (if
the Initial Developer is not the Participant) and all Contributors
under Sections 2.1 and/or 2.2 of this License shall, upon 60 days
notice from Participant terminate prospectively and automatically
at the expiration of such 60 day notice period, unless if within
such 60 day period You withdraw Your claim with respect to the
Participant Software against such Participant either unilaterally
or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant
alleging that the Participant Software directly or indirectly
infringes any patent where such claim is resolved (such as by
license or settlement) prior to the initiation of patent
infringement litigation, then the reasonable value of the licenses
granted by such Participant under Sections 2.1 or 2.2 shall be
taken into account in determining the amount or value of any
payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above,
all end user licenses that have been validly granted by You or any
distributor hereunder prior to termination (excluding licenses
granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT
(INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE
INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF
COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE
LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT
LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER
FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR
LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE
POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL
NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM
SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS
SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR
LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS
EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.
8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
1.1023 nocache 1.0.1 1.0.1

1.1023.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1024 path-parse 1.0.6 1.0.6

1.1024.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Javier Blanco

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1025 @babel/plugin-syntax-export-namespace-from 7.8.3
1.1025.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1026 hive 3.1.2
1.1026.1 Available under license :
Javolution - Java(tm) Solution for Real-Time and Embedded Systems
Copyright (c) 2012, Javolution (http://javolution.org/)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright  2002, Steve Souza (admin@jamonapi.com)
All rights reserved.
Modifications: No
Redistribution in binary form, with or without modifications, are permitted provided that the following conditions are met:

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* If modifications are made to source code then this license should indicate that fact in the "Modifications" section above.

* Neither the author, nor the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

**THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.**

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.
1.5. "Incompatible With Secondary Licenses" means
(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form" means any form of the work other than Source Code Form.

1.7. "Larger Work" means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License" means this document.

1.9. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications" means any of the following:
(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or
(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License" means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions
-----------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software;
(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not
3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support,
indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation
-----------------------------------------------

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Software due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description must
be placed in a text file included with all distributions of the Covered
Software under this License. Except to the extent prohibited by statute
or regulation, such description must be sufficiently detailed for a
recipient of ordinary skill to be able to understand it.

5. Termination
-------

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

**************************************************************************
|                                                                      |
**************************************************************************
* 6. Disclaimer of Warranty
*  
* Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

* 7. Limitation of Liability
*  
* Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If you choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as
defined by the Mozilla Public License, v. 2.0.
Binary distributions of this software contain jars that are not licensed under the
Apache License 2.0. Additional licenses attached to these jars are contained in the
same directory as this NOTICE file.

Note also that some of the included jars impose additional restrictions on you if
you modify the source code of those jars. In particular see:

* https://glassfish.dev.java.net/public/CDDLv1.0.html for information on the CDDL
license that glassfish is distributed under and

* http://www.mozilla.org/MPL/MPL-1.1.txt for information on the Mozilla Public License
which jamon-runtime and rhino are distributed under.

* https://opensource.org/licenses/cddl1.php for information on CDDL 1.0 which
transaction-api is distributed under.

Copyright (c) 2001-2016, Arno Unkrig
Copyright (c) 2015-2016  TIBCO Software Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials
provided with the distribution.
3. Neither the name of JANINO nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2006 Paul Hammant & ThoughtWorks Inc
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2012 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

* Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0 1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following: A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications; B. Any new file that contains any part of the Original Software or previous Modification; or C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by
the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY. COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.
6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS. The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS. This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS. As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal
Binary distributions of this software contain jars that are not licensed under the Apache License 2.0. Additional licenses attached to these jars are contained in the same directory as this NOTICE file.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2011, abego Software GmbH, Germany (http://www.abego.org)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the abego Software GmbH nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Apache Hive
Copyright 2008-2019 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This project includes software licensed under the JSON license.

Apache License
Version 2.0, January 2017
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HIVE SUBCOMPONENTS:

The Apache Hive project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

For the SQLLine package:

Copyright (c) 2002, 2003, 2004, 2005 Marc Prud'hommeaux

From: http://sqlline.sourceforge.net/#license

"SQLLine is distributed under the BSD License, meaning that you are free to redistribute, modify, or sell the software with almost no restrictions."

Statement from Marc Prud'hommeaux regarding inconsistent licenses in some SQLLine source files:

> SQLLine was once GPL, but it was changed to be BSD a few years back.
> Any references to the GPL are vestigial. Hopefully the license declaration at http://sqlline.sourceforge.net/#license is sufficiently authoritative in this regard.

For the org.apache.hive.beeline.ClassNameCompleter class:

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For org.apache.hadoop.hive.llap.daemon.impl.PriorityBlockingDeque class:

The BSD 3-Clause License

Copyright (c) 2007, Aviad Ben Dov

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Infomancers, Ltd. nor the names of its contributors may be
used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For jquery.sparkline.js:

License: New BSD License (3-clause)

Copyright (c) 2012, Splunk Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Splunk Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, 
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For json.human.js/json.human.css:

Copyright (c) 2016 Mariano Guerra

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For argparse.py:

argparse is (c) 2006-2009 Steven J. Bethard <steven.bethard@gmail.com>.

The argparse module was contributed to Python as of Python 2.7 and thus was licensed under the Python license. Same license applies to all files in the argparse package project.

For details about the Python License, please see doc/Python-License.txt.

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce,
analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement. This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2008-2017 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2016 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.


# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice,
# this list of conditions and the following disclaimer.
# Redistributions in binary form must reproduce the above
# copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided with
# the distribution.
# Neither the name of Google Inc. nor the names of its
# contributors may be used to endorse or promote products derived from
# this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
The word list in cjdict.txt are generated by combining three word lists listed below with further processing for compound word breaking. The frequency is generated with an iterative training against Google web corpora.

* Libtabe (Chinese)
  - https://sourceforge.net/project/?group_id=1519
  - Its license terms and conditions are shown below.

* IPADIC (Japanese)
  - http://chasen.aist-nara.ac.jp/chasen/distribution.html
  - Its license terms and conditions are shown below.

---------COPYING.libtabe ---- BEGIN-------------------

/*
 * Copyright (c) 1999 TaBE Project.
 * Copyright (c) 1999 Pai-Hsiang Hsiao.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * 
 * . Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * . Neither the name of the TaBE Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
 * 
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

Open Source Used In DNAC Platform Arica 3594
* REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* */

/*

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.
. Neither the name of the Computer Systems and Communication Lab
nor the names of its contributors may be used to endorse or
promote products derived from this software without specific
prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
*(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)*
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)*
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
University of Illinois
c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4

# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# to users as so produced on an experimental basis. Accordingly, the
# program is provided without any warranty whatsoever, whether express,
# implied, statutory or otherwise. The term "warranty" used herein
# includes, but is not limited to, any warranty of the quality,
# performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
# Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
“use” as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.

In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.

This file is derived from the above dictionary, with slight
modifications.

Copyright (c) 2013 Brian Eugene Wilson, Robert Martin Campbell.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification,
# are permitted provided that the following conditions are met:
#    
# Redistribution of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in
# binary form must reproduce the above copyright notice, this list of
# conditions and the following disclaimer in the documentation and/or
# other materials provided with the distribution.
#    
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
#    
4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#    
# This list is part of a project hosted at:
#    github.com/kanyawtech/myanmar-karen-word-lists
#    
# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#    
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
# disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided
# with the distribution.
#    
# Neither the name Myanmar Karen Word Lists, nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
# BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
# EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
# TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE;
# DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
# ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
# TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
# THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
# SUCH DAMAGE.
# ----------------------------------------------------------------------

5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

# 7. Database Ownership

# The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

Copyright (c) 2002,2003,2004,2005,2006 Marc Prud'hommeaux
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2012 Terence Parr and Sam Harwell
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
Apache Hive Metastore

Copyright 2008-2017 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE Version 1.0 (CDDL-1.0)

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor
(if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by
the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial
grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing
Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally
released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation,
method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2004-2013 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2003-2008, Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

1.1027.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1028 metrics-json 4.1.0

1.1029 findbugs-annotations 1.3.9-1

1.1030.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1031 copy-to-clipboard 3.3.1 3.3.1
1.1031.1 Available under license:

MIT License

Copyright (c) 2017 sudodoki <smd.deluzion@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1032 commons-validator 1.6

1.1032.1 Available under license:

Apache Commons Validator
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1033 jackson-databind 2.4.0

1.1033.1 Available under license :

No license file was found, but licenses were detected in source scan.

<p class="legalCopy"><small>Copyright &amp;#169; 2014 &lt;a href="http://fasterxml.com/"&gt;FasterXML.&lt;/a&gt; . All Rights Reserved. &lt;/small&gt;&lt;/p&gt;
<table>
<thead>
<tr>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/JsonDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/jsontype/impl/TypeNameIdResolver.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/class-use/SettableAnyProperty.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/ContextualKeyDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/cfg/package-use.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/ser/std/class-use/StringSerializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/node/package-summary.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/introspect/class-use/AnnotatedConstructor.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/class-use/ContextualDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/annotation/class-use/JsonTypeIdResolver.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/std/NumberDeserializers.BooleanDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/std/class-use/AtomicBooleanDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/class-use/BeanDescription.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/node/NullNode.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/jsontype/TypeResolverBuilder.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/ser/std/class-use/RawSerializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/introspect/WithMember.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/jsontype/class-use/SchemaAware.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/SettableAnyProperty.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/std/class-use/EnumDeserializers.FactoryBasedDeserializer.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/deser/std/NumberSerializers.Base.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/cfg/ContextAttributes.html</td>
</tr>
<tr>
<td>* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode-</td>
</tr>
<tr>
<td>jar/com/fasterxml/jackson/databind/type/ClassKey.html</td>
</tr>
</tbody>
</table>
Open Source Used in DNAC Platform Arica 3628

* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/json/type/NamedType.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/util/BeanUtil.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/ser/std/class-use/EnumMapSerializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/deser/impl/BeanPropertyMap.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/deser/impl/ValueInjector.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/deser/SettableBeanProperty.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/ser/std/class-use/FilterProvider.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/deser/std/class-use/DateDeserializers.DateDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/node/IntNode.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/ser/impl/class-use/IteratorSerializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/deser/std/class-use/NumberDeserializers.ShortDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacode.jar/com/fasterxml/jackson/databind/ext/class-use/DOMDeserializer.html
Open Source Used In DNAC Platform Arica 3637
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/set/SerializerFactory.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/jsontype/impl/class-use/TypeIdResolverBase.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/jsontype/impl/class-use/AsArrayTypeDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/ser/std/class-use/NumberSerializers.IntLikeSerializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/impl/ExternalTypeHandler.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/jsontype/class-use/TypeIdResolver.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/introspect/AnnotatedMethod.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/DateDeserializers.TimestampDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/ResolvableDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/package-summary.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/StdArraySerializers.TypedReader.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/class-use/NumberDeserializers.FloatDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/NumberDeserializers.FloatDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/NumberDeserializers.BigDecimalDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/NumberDeserializers.BigDecimalDeserializer.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/package-summary.html
* /opt/cola/permits/1028374027_1609784616.87/0/jackson-databind-2-4-0-javacod-jar/com/fasterxml/jackson/databind/deser/std/package-summary.html
Open Source Used In DNAC Platform Artica 3643
No license file was found, but licenses were detected in source scan.

Since 2.4 there are special exceptions for JSON Tree model

**1.1034 reap2 1.0.1 1.0.1**
### 1.1034.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1035 crypto-random-string 2.0.0

#### 1.1035.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1036 argparse 1.0.10 1.0.10
1.1036.1 Available under license:
(The MIT License)

Copyright (C) 2012 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1037 @jsreport/mingo 2.4.1 2.4.1

1.1037.1 Available under license:
The MIT License (MIT)

Copyright (c) 2017 Francis Asante

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1038 commander 4.1.1 4.1.1
1.1038.1 Available under license :
(The MIT License)

Copyright (c) 2011 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1039 javolution 5.5.1
1.1039.1 Available under license :
No license file was found, but licenses were detected in source scan.

/**
 * Interface for an element's attribute specifications.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * This is the original SAX1 interface for reporting an element's
 * attributes. Unlike the new [{link xml.sax.Attributes} Attributes] interface, it does not support Namespace-related information.</p>
 *
When an attribute list is supplied as part of a
[ @link org.xml.sax.DocumentHandler#startElement startElement] event, the list will return valid results only during the
scope of the event; once the event handler returns control
to the parser, the attribute list is invalid. To save a
persistent copy of the attribute list, use the SAX1 AttributeListImpl
helper class.</p>

An attribute list includes only attributes that have been
specified or defaulted: #IMPLIED attributes will not be included.</p>

There are two ways for the SAX application to obtain information
from the AttributeList. First, it can iterate through the entire
list:</p>

<pre>
public void startElement (String name, AttributeList atts) {
  for (int i = 0; i < atts.getLength(); i++) {
    String name = atts.getName(i);
    String type = atts.getType(i);
    String value = atts.getValue(i);
    [...]
  }
}
</pre>

(Note that the result of getLength() will be zero if there
are no attributes.)

As an alternative, the application can request the value or
type of specific attributes:</p>

<pre>
public void startElement (String name, AttributeList atts) {
  String identifier = atts.getValue("id");
  String label = atts.getValue("label");
  [...]
}
</pre>

@since SAX 1.0
@author David Megginson
@version 2.0.1 (sax2r2)
No license file was found, but licenses were detected in source scan.

/**
 * Receive notification of general document events.
 *
 * <blockquote>
 * This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. See
 * [http://www.saxproject.org](http://www.saxproject.org) for further information.
 * </blockquote>
 *
 * This was the main event-handling interface for SAX1; in
 * SAX2, it has been replaced by `ContentHandler`, which provides Namespace support and reporting
 * of skipped entities. This interface is included in SAX2 only
 * to support legacy SAX1 applications.
 *
 * The order of events in this interface is very important, and
 * mirrors the order of information in the document itself. For
 * example, all of an element's content (character data, processing
 * instructions, and/or subelements) will appear, in order, between
 * the startElement event and the corresponding endElement event.
 *
 * Application writers who do not want to implement the entire
 * interface can derive a class from `HandlerBase`, which implements
 * the default functionality; parser writers can instantiate
 * `HandlerBase` to obtain a default handler. The application can find
 * the location of any document event using the `Locator` interface
 * supplied by the Parser through the `setDocumentLocator` method.
 *
 * @since SAX 1.0
 * @author David Megginson
 * @version 2.0.1 (sax2r2)
 * @see org.xml.sax.Parser#setDocumentHandler
 * @see org.xml.sax.Locator
 * @see org.xml.sax.HandlerBase
 */

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javadoc/javadoc-sources-2-
  jar/org/xml/sax/DocumentHandler.java

No license file was found, but licenses were detected in source scan.

/**
 * Basic interface for resolving entities.
 *
 * <blockquote>
* This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.<br/>
* See [http://www.saxproject.org](http://www.saxproject.org) for further information.

* If a SAX application needs to implement customized handling for external entities, it must implement this interface and register an instance with the SAX driver using the `@link org.xml.sax.XMLReader#setEntityResolver setEntityResolver` method.

* The XML reader will then allow the application to intercept any external entities (including the external DTD subset and external parameter entities, if any) before including them.

* Many SAX applications will not need to implement this interface, but it will be especially useful for applications that build XML documents from databases or other specialised input sources, or for applications that use URI types other than URLs.

* The following resolver would provide the application with a special character stream for the entity with the system identifier "http://www.myhost.com/today":

```java
import org.xml.sax.EntityResolver;
import org.xml.sax.InputSource;

public class MyResolver implements EntityResolver {
    public InputSource resolveEntity (String publicId, String systemId)
    {
        if (systemId.equals("http://www.myhost.com/today")) {
            // return a special input source
            MyReader reader = new MyReader();
            return new InputSource(reader);
        } else {
            // use the default behaviour
            return null;
        }
    }
}
```

* The application can also use this interface to redirect system identifiers to local URIs or to look up replacements in a catalog (possibly by using the public identifier).
/**
 * Interface for associating a SAX event with a document location.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. </em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * If a SAX parser provides location information to the SAX application, it does so by implementing this interface and then
 * passing an instance to the application using the content handler's [ @link org.xml.sax.ContentHandler#setDocumentLocator
 * setDocumentLocator] method. The application can use the object to obtain the location of any other SAX event
 * in the XML source document.</p>
 *
 * Note that the results returned by the object will be valid only during the scope of each callback method: the application
 * will receive unpredictable results if it attempts to use the locator at any other time, or after parsing completes.</p>
 *
 * SAX parsers are not required to supply a locator, but they are very strongly encouraged to do so. If the parser supplies a
 * locator, it must do so before reporting any other document events. If no locator has been set by the time the application receives
 * the [ @link org.xml.sax.ContentHandler#startDocument startDocument] event, the application should assume that a locator is not
 * available.</p>
 *
 * @since SAX 1.0
 * @author David Megginson
 * @version 2.0.1 (sax2r2)
 * @see org.xml.sax.ContentHandler#setDocumentLocator
 */

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/org/xml/sax/EntityResolver.java

No license file was found, but licenses were detected in source scan.
// NO WARRANTY! This class is in the Public Domain.
/**
 * Exception class for an unsupported operation.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>. </em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * @since SAX 2.0
 * @author David Megginson
 * @version 2.0.1 (sax2r2)
 * @see org.xml.sax.SAXNotRecognizedException
 */
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/Serializable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/lang/Immutable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/Formattable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javax/realtime/RealtimeThread.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/logging/Logger.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/FormatFlagsConversionMismatchException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/Comparable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/TooManyListenersException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/text/Text.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/NoSuchFieldException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/io/Union.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/Queue.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/NotActiveException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/InvalidClassException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/io/Union.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/IllegalFormatPrecisionException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/Configurable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/java/util/Set.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/NoSuchFieldException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/io/Union.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/Queue.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputStream.java
<table>
<thead>
<tr>
<th>File Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jar/templates/javolution/util/FastList.java</td>
<td>File containing Java code for the FastList class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/text/ParsePosition.java</td>
<td>File containing Java code for the ParsePosition class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/io/NotSerializableException.java</td>
<td>File containing Java code for the NotSerializableException class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/LinkedHashMap.java</td>
<td>File containing Java code for the LinkedHashMap class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/logging/Level.java</td>
<td>File containing Java code for the Level class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/nio/ByteOrder.java</td>
<td>File containing Java code for the ByteOrder class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/lang/Number.java</td>
<td>File containing Java code for the Number class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/ListIterator.java</td>
<td>File containing Java code for the ListIterator class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/nio/InvalidMarkException.java</td>
<td>File containing Java code for the InvalidMarkException class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/SortedSet.java</td>
<td>File containing Java code for the SortedSet class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/nio/UTF8ByteBufferReader.java</td>
<td>File containing Java code for the UTF8ByteBufferReader class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/io/ObjectInput.java</td>
<td>File containing Java code for the ObjectInput class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/Observable.java</td>
<td>File containing Java code for the Observable class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/lang/ThreadLocal.java</td>
<td>File containing Java code for the ThreadLocal class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/ArrayList.java</td>
<td>File containing Java code for the ArrayList class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/InputMismatchException.java</td>
<td>File containing Java code for the InputMismatchException class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/UnknownFormatFlagsException.java</td>
<td>File containing Java code for the UnknownFormatFlagsException class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/javolution/xml/sax/XMLReaderImpl.java</td>
<td>File containing Java code for the XMLReaderImpl class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/nio/ByteBuffer.java</td>
<td>File containing Java code for the ByteBuffer class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/templates/java/util/SortedSet.java</td>
<td>File containing Java code for the SortedSet class.</td>
</tr>
<tr>
<td>/opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/util/Index.java</td>
<td>File containing Java code for the Index class.</td>
</tr>
</tbody>
</table>
jar/_templates/java/util/ConcurrentModificationException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/text/CharArray.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/util/concurrent/ConcurrentMap.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectInputStream.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutputValidation.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/ObjectOutput.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/xml/stream/XMLStreamException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/IllegalStateException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/io/Union.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/lang/CloneNotSupportedException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/java/io/SyncFailedException.java
No license file was found, but licenses were detected in source scan.

/*
 * Javolution - Java(TM) Solution for Real-Time and Embedded Systems
 * Copyright (C) 2009 - Javolution (http://javolution.org/)
 * All rights reserved.
 */
Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

Found in path(s):
- /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/context/PoolContext.java
- /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/context/PoolContext.java

No license file was found, but licenses were detected in source scan.

Basic interface for SAX (Simple API for XML) parsers.

This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.


This was the main event supplier interface for SAX1; it has been replaced in SAX2 by XMLReader, which includes Namespace support and sophisticated configurability and extensibility.

All SAX1 parsers must implement this basic interface: it allows applications to register handlers for different types of events and to initiate a parse from a URI, or a character stream.

All SAX1 parsers must also implement a zero-argument constructor (though other constructors are also allowed).

SAX1 parsers are reusable but not re-entrant: the application may reuse a parser object (possibly with a different input source) once the first parse has completed successfully, but it may not invoke the parse() methods recursively within a parse.

@deprecated This interface has been replaced by the SAX2 interface, which includes Namespace support.

@author David Megginson
@version 2.0.1 (sax2r2)
@see org.xml.sax.EntityResolver
@see org.xml.sax.DTDHandler
@see org.xml.sax.DocumentHandler
/**
 * Encapsulate an XML parse error or warning.
 *
 * <blockquote>
 * This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>. See <a href='http://www.saxproject.org'>http://www.saxproject.org</a> for further information.
 * </blockquote>
 *
 * This exception may include information for locating the error in the original XML document, as if it came from a Locator object. Note that although the application will receive a SAXParseException as the argument to the handlers in the ErrorHandler interface, the application is not actually required to throw the exception; instead, it can simply read the information in it and take a different action.</p>
 *
 * Since this exception is a subclass of SAXException, it inherits the ability to wrap another exception.</p>
 *
 * @since SAX 1.0
 * @author David Megginson
 *
 * @see org.xml.sax.SAXException
 * @see org.xml.sax.Locator
 * @see org.xml.sax.ErrorHandler
 */

// NO WARRANTY! This class is in the Public Domain.
/**
 * Interface for an XML filter.
 *
 */
This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY.<br/>
See [http://www.saxproject.org](http://www.saxproject.org) for further information.

An XML filter is like an XML reader, except that it obtains its events from another XML reader rather than a primary source like an XML document or database. Filters can modify a stream of events as they pass on to the final application.

The XMLFilterImpl helper class provides a convenient base for creating SAX2 filters, by passing on all EntityResolver, DTDHandler, ContentHandler and ErrorHandler events automatically.

* @since SAX 2.0
* @author David Megginson
* @version 2.0.1 (sax2r2)
*/

No license file was found, but licenses were detected in source scan.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (C) 2006 - Javolution (http://javolution.org/)
All rights reserved.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (C) 2006 - Javolution (http://javolution.org/)
All rights reserved.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.
/**
 * Receive notification of basic DTD-related events.
 *
 * <em>This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY</em>.
 *
 * If a SAX application needs information about notations and unparsed entities, then the application implements this interface and registers an instance with the SAX parser using the parser's setDTDHandler method. The parser uses the instance to report notation and unparsed entity declarations to the application.
 *
 * Note that this interface includes only those DTD events that the XML recommendation requires processors to report:
 * notation and unparsed entity declarations.
 *
 * The SAX parser may report these events in any order, regardless of the order in which the notations and unparsed entities were declared; however, all DTD events must be reported after the document handler's startDocument event, and before the first startElement event.
 * (If the LexicalHandler is used, these events must also be reported before the endDTD event.)
 */
It is up to the application to store the information for future use (perhaps in a hash table or object tree).

If the application encounters attributes of type "NOTATION", "ENTITY", or "ENTITIES", it can use the information that it obtained through this interface to find the entity and/or notation corresponding with the attribute value.

@since SAX 1.0
@author David Megginson
@version 2.0.1 (sax2r2)

No license file was found, but licenses were detected in source scan.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (C) 2008 - Javolution (http://javolution.org/)
All rights reserved.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

No license file was found, but licenses were detected in source scan.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (C) 2009 - Javolution (http://javolution.org/)
All rights reserved.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

No license file was found, but licenses were detected in source scan.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (C) 2009 - Javolution (http://javolution.org/)
All rights reserved.

Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
 * Interface for a list of XML attributes.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * This interface allows access to a list of attributes in
 * three different ways:
 *
 * <ol>
 * <li>by attribute index;</li>
 * <li>by Namespace-qualified name; or</li>
 * <li>by qualified (prefixed) name.</li>
 * </ol>
 *
 * The list will not contain attributes that were declared
 * #IMPLIED but not specified in the start tag. It will also not
 * contain attributes used as Namespace declarations (xmlns*) unless
 * the <code>http://xml.org/sax/features/namespace-prefixes</code>
 * feature is set to <var>true</var> (it is <var>false</var> by
 * default).
 * Because SAX2 conforms to the original "Namespaces in XML"
 * recommendation, it normally does not
 * give namespace declaration attributes a namespace URI.
 *
 * Some SAX2 parsers may support using an optional feature flag
 * (<code>http://xml.org/sax/features/xmlns-uris</code>) to request
 * that those attributes be given URIs, conforming to a later
 * backwards-incompatible revision of that recommendation. (The
 * attribute's "local name" will be the prefix, or "xmlns" when
 * defining a default element namespace.) For portability, handler
 * code should always resolve that conflict, rather than requiring
 * parsers that can change the setting of that feature flag. </p>
 *
 * If the namespace-prefixes feature (see above) is
 * <var>false</var>, access by qualified name may not be available; if
 * the <code>http://xml.org/sax/features/namespaces</code> feature is
 * <var>false</var>, access by Namespace-qualified names may not be
 * available.}</p>
This interface replaces the now-deprecated SAX1 `AttributeList` interface, which does not contain Namespace support. In addition to Namespace support, it adds the `getIndex` methods (below).

The order of attributes in the list is unspecified, and will vary from implementation to implementation.

@since SAX 2.0
@since SAX 1.0
@version 2.0.1 (sax2r2)

Encapsulate a general SAX error or warning.

This class can contain basic error or warning information from either the XML parser or the application: a parser writer or application writer can subclass it to provide additional functionality. SAX handlers may throw this exception or any exception subclassed from it.

If the application needs to pass through other types of exceptions, it must wrap those exceptions in a SAXException or an exception derived from a SAXException.

If the parser or application needs to include information about a specific location in an XML document, it should use the [SAXParseException](http://www.saxproject.org) subclass.

Encapsulate a general SAX error or warning.

This class can contain basic error or warning information from either the XML parser or the application: a parser writer or application writer can subclass it to provide additional functionality. SAX handlers may throw this exception or any exception subclassed from it.

If the application needs to pass through other types of exceptions, it must wrap those exceptions in a SAXException or an exception derived from a SAXException.

If the parser or application needs to include information about a specific location in an XML document, it should use the [SAXParseException](http://www.saxproject.org) subclass.
// NO WARRANTY! This class is in the Public Domain.
/**
 * Interface for reading an XML document using callbacks.
 *<blockquote>
 * <em>This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>. See <a href='http://www.saxproject.org'>http://www.saxproject.org</a> for further information.</em>
 *</blockquote>
 *
 * Note: despite its name, this interface does not extend the standard Java Reader interface, because reading XML is a fundamentally different activity than reading character data.
 *
 * XMLReader is the interface that an XML parser's SAX2 driver must implement. This interface allows an application to set and query features and properties in the parser, to register event handlers for document processing, and to initiate a document parse.
 *
 * All SAX interfaces are assumed to be synchronous: the methods must not return until parsing is complete, and readers must wait for an event-handler callback to return before reporting the next event.
 *
 * This interface replaces the (now deprecated) SAX 1.0 interface. The XMLReader interface contains two important enhancements over the old Parser interface (as well as some minor ones):
 *
 * it adds a standard way to query and set features and properties; and
 * it adds Namespace support, which is required for many higher-level XML standards.
 *
 * There are adapters available to convert a SAX1 Parser to a SAX2 XMLReader and vice-versa.
 *
 * @since SAX 2.0
 * @author David Megginson
Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/org/xml/sax/XMLReader.java
No license file was found, but licenses were detected in source scan.

/**
 * Javolution - Java(TM) Solution for Real-Time and Embedded Systems
 * Copyright (C) 2006 - Javolution (http://javolution.org/)
 * All rights reserved.
 * Permission to use, copy, modify, and distribute this software is
 * freely granted, provided that this notice is preserved.
 */

/**
 * ====================================================
 * Copyright (C) 1993 by Sun Microsystems, Inc. All rights reserved.
 * Developed at SunSoft, a Sun Microsystems, Inc. business.
 * Permission to use, copy, modify, and distribute this
 * software is freely granted, provided that this notice
 * is preserved.
 * ====================================================
 */

/**
 * ====================================================
 * Copyright (C) 2004 by Sun Microsystems, Inc. All rights reserved.
 * Permission to use, copy, modify, and distribute this
 * software is freely granted, provided that this notice
 * is preserved.
 * ====================================================
 */

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/lang/MathLib.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/lang/MathLib.java
No license file was found, but licenses were detected in source scan.

/**
 * Default base class for handlers.
 *<blockquote>
 *<em>This module, both source code and documentation, is in the
 */
* This class implements the default behaviour for four SAX1 interfaces: EntityResolver, DTDHandler, DocumentHandler, and ErrorHandler. It is now obsolete, but is included in SAX2 to support legacy SAX1 applications. SAX2 applications should use the DefaultHandler class instead.</p>

* Application writers can extend this class when they need to implement only part of an interface; parser writers can instantiate this class to provide default handlers when the application has not supplied its own.</p>

* Note that the use of this class is optional.

* @deprecated This class works with the deprecated DocumentHandler interface. It has been replaced by the SAX2 DefaultHandler class.

* @since SAX 1.0
* @author David Megginson
* @version 2.0.1 (sax2r2)
* @see org.xml.sax.EntityResolver
* @see org.xml.sax.DTDHandler
* @see org.xml.sax.DocumentHandler
* @see org.xml.sax.ErrorHandler
*/

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.560/javolution-5-5-1-sources-2-jar/org/xml/sax/HandlerBase.java

No license file was found, but licenses were detected in source scan.

/**
* Basic interface for SAX error handlers.
*
* <blockquote>
* This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>. See <a href="http://www.saxproject.org">http://www.saxproject.org</a> for further information.
* </blockquote>

* If a SAX application needs to implement customized error handling, it must implement this interface and then register an instance with the XML reader using the
* { @link org.xml.sax.XMLReader#setErrorHandler setErrorHandler }

---

Open Source Used In DNAC Platform Arica 3685
<p>* method. The parser will then report all errors and warnings through this interface.</p>

*<p>* <strong>WARNING:</strong> If an application does <em>not</em> register an ErrorHandler, XML parsing errors will go unreported, except that <em>SAXParseException</em>s will be thrown for fatal errors. In order to detect validity errors, an ErrorHandler that does something with <em>{@link #error error()}</em> calls must be registered.</p>

*<p>* For XML processing errors, a SAX driver must use this interface in preference to throwing an exception: it is up to the application to decide whether to throw an exception for different types of errors and warnings. Note, however, that there is no requirement that the parser continue to report additional errors after a call to <em>{@link #fatalError fatalError}</em>. In other words, a SAX driver class may throw an exception after reporting any fatalError. Also parsers may throw appropriate exceptions for non-XML errors. For example, <em>{@link XMLReader#parse XMLReader.parse()}</em> would throw an IOException for errors accessing entities or the document.</p>

*<p>* @since SAX 1.0
* @author David Megginson
* @version 2.0.1+ (sax2r3pre1)
* @see org.xml.sax.XMLReader#setErrorHandler
* @see org.xml.sax.SAXParseException
*/

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/org/xml/sax/ErrorHandler.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the Public Domain.
/**
* Exception class for an unrecognized identifier.
*<p>* <em>This module, both source code and documentation, is in the Public Domain, and comes with <strong>NO WARRANTY</strong>.</em></p>
*/

*<p>* An XMLReader will throw this exception when it finds an unrecognized feature or property identifier; SAX applications and extensions may use this class for other, similar purposes.</p>

*<p>* @since SAX 2.0
* @author David Megginson
* @version 2.0.1 (sax2r2)
* @see org.xml.sax.SAXNotSupportedException
*/

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-
  jar/org/xml/sax/SAXNotRecognizedException.java
No license file was found, but licenses were detected in source scan.

/**
 * A single input source for an XML entity.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with <strong>NO WARRANTY</strong>.<em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * <p>This class allows a SAX application to encapsulate information
 * about an input source in a single object, which may include
 * a public identifier, a system identifier, a byte stream (possibly
 * with a specified encoding), and/or a character stream.</p>
 *
 * <p>There are two places that the application can deliver an
 * input source to the parser: as the argument to the Parser.parse
 * method, or as the return value of the EntityResolver.resolveEntity
 * method.</p>
 *
 * <p>The SAX parser will use the InputSource object to determine how
 * to read XML input. If there is a character stream available, the
 * parser will read that stream directly, disregarding any text
 * encoding declaration found in that stream.
 * If there is no character stream, but there is
 * a byte stream, the parser will use that byte stream, using the
 * encoding specified in the InputSource or else (if no encoding is
 * specified) autodetecting the character encoding using an algorithm
 * such as the one in the XML specification. If neither a character
 * stream nor a
 * byte stream is available, the parser will attempt to open a URI
 * connection to the resource identified by the system
 * identifier.</p>
 *
 * <p>An InputSource object belongs to the application: the SAX parser
 * shall never modify it in any way (it may modify a copy if
 * necessary). However, standard processing of both byte and
 * character streams is to close them on as part of end-of-parse cleanup,
 * so applications should not attempt to re-use such streams after they
Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/org/xml/sax/InputSource.java
No license file was found, but licenses were detected in source scan.

// NO WARRANTY! This class is in the public domain.
/**
 * Receive notification of the logical content of a document.
 *
 * <blockquote>
 * <em>This module, both source code and documentation, is in the
 * Public Domain, and comes with *strong>NO WARRANTY</strong>.</em>
 * See <a href='http://www.saxproject.org'>http://www.saxproject.org</a>
 * for further information.
 * </blockquote>
 *
 * This is the main interface that most SAX applications
 * implement: if the application needs to be informed of basic parsing
 * events, it implements this interface and registers an instance with
 * the SAX parser using the {@link org.xml.sax.XMLReader#setContentHandler
 * setContentHandler} method. The parser uses the instance to report
 * basic document-related events like the start and end of elements
 * and character data.</p>
 *
 * The order of events in this interface is very important, and
 * mirrors the order of information in the document itself. For
 * example, all of an element's content (character data, processing
 * instructions, and/or subelements) will appear, in order, between
 * the startElement event and the corresponding endElement event.</p>
 *
 * This interface is similar to the now-deprecated SAX 1.0
 * DocumentHandler interface, but it adds support for Namespaces
 * and for reporting skipped entities (in non-validating XML
 * processors).</p>
 *
 * Implementors should note that there is also a
 * <code>ContentHandler</code> class in the <code>java.net</code>
 * package; that means that it's probably a bad idea to do</p>
 *
 * import java.net.*;
* import org.xml.sax.*;
* </pre>
*
* <p>In fact, "import ...*" is usually a sign of sloppy programming anyway, so the user should consider this a feature rather than a bug.</p>
*
* @since SAX 2.0
* @author David Megginson
* @version 2.0.1+ (sax2r3pre1)
* @see org.xml.sax.XMLReader
* @see org.xml.sax.DTDHandler
* @see org.xml.sax.ErrorHandler
*/

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/org/xml/sax/ContentHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Javolution - Java(TM) Solution for Real-Time and Embedded Systems
* Copyright (C) 2007 - Javolution (http://javolution.org/)
* All rights reserved.
* Permission to use, copy, modify, and distribute this software is
* freely granted, provided that this notice is preserved.
*/

Found in path(s):
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/testing/AssertionException.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/xml/XMLSerializable.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/xml/QName.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/context/Context.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/context/ArrayFactory.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/xml/ws/WebServiceClient.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/javolution/context/ImmortalContext.java
* /opt/cola/permits/1118987831_1608672578.56/0/javolution-5-5-1-sources-2-jar/_templates/javolution/testing/TestCase.java
1.1040 isobject 2.1.0

1.1040.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1041 compression 1.6.2 1.6.2
1.1041.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1042 to-readable-stream 1.0.0
1.1042.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1043 is-stream 1.1.0 1.1.0

1.1043.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1044 antlr 3.1.3

1.1044.1 Available under license:

[The "BSD licence"]
Copyright (c) 2008 Erik van Bilsen
Copyright (c) 2005-2007 Kunle Odutola
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code MUST RETAIN the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form MUST REPRODUCE the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior WRITTEN permission.
4. Unless explicitly state otherwise, any contribution intentionally submitted for inclusion in this work to the copyright owner shall be under the terms and conditions of this license, without any additional terms or conditions.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ANTLR 3 License

[The BSD License]
Copyright (c) 2010 Terence Parr
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived...
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[The "BSD licence"]
Copyright (c) 2003-2006 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[The "BSD licence"]
Copyright (c) 2005-2007 Kunle Odutola
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code MUST RETAIN the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form MUST REPRODUCE the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior WRITTEN permission.
4. Unless explicitly state otherwise, any contribution intentionally submitted for inclusion in this work to the copyright owner shall be under the terms and conditions of this license, without any additional terms or conditions.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// [The "BSD licence"]
// Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
// http://www.temporal-wave.com
// http://www.linkedin.com/in/jimidle
//
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
// 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
[The "BSD licence"]
Copyright (c) 2007-2008 Leon, Jen-Yuan Su
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
/*
Copyright (c) 2003-2008 Terence Parr. All rights reserved.
Code licensed under the BSD License:
http://www.antlr.org/license.html

Some parts of the ANTLR class:
Copyright (c) 2008, Yahoo! Inc. All rights reserved.
Code licensed under the BSD License:
http://developer.yahoo.net/yui/license.txt
*/

1.1045 has 1.0.3 1.0.3

1.1045.1 Available under license:
Copyright (c) 2013 Thiago de Arruda

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.1046 ratelimitj-core 0.4.0
1.1046.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1047 lz4-java 1.7.1

1.1047.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1048 bluebird 3.5.1 3.5.1
1.1048.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013-2017 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1049 mime-pull 1.9.3

1.1050 drill-opentsdb-storage 1.19.0
1.1050.1 Available under license:

Drill : Contrib : Storage : OpenTSDB
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
1.1051 okio 2.4.3
1.1051.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications
or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the
Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright
owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written
communication sent to the Licensor or its representatives, including but not limited to communication on electronic
mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously
marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been
received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section)
patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent
infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date
such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any
medium, with or without modifications, and in Source or Object form, provided that You meet the following
conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and
   You must cause any modified files to carry prominent notices stating that You changed the files; and
   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
   and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
   the Derivative Works; and
   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You
distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding
those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a
NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided
along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such
third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not
modify the License. You may add Your own attribution notices within Derivative Works that You distribute,
alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices
cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1052 babel-plugin-syntax-object-rest-spread 6.13.0

1.1052.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Open Source Used In DNAC Platform Arica 3716

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1053 socket.io-client 1.7.4 1.7.4
1.1053.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Guillermo Rauch

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1054 jsreport-html-to-xlsx 2.8.3 2.8.3

1.1054.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.
An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would otherwise
be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the
1.1055 babel-traverse 6.26.0

1.1055.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1056 de.huxhorn.sulky.codec 0.9.17

1.1056.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.
The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

---

1.1057 jersey-media-jaxb 2.32

1.1057.1 Available under license:

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc
Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.
org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity.
Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient’s receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients’ rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements
of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance
claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost
of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.1058 yargs-parser 13.1.2 13.1.2

1.1058.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1059 vary 1.0.1 1.0.1
1.1059.1 Available under license:
(The MIT License)

Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1060 drill-yarn 1.19.0
1.1060.1 Available under license:

Drill : On-YARN
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1061 async-limiter 1.0.1 1.0.1

1.1061.1 Available under license:

The MIT License (MIT)
Copyright (c) 2017 Samuel Reed <samuel.trace.reed@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1062 jnr-netdb 1.1.6

1.1062.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 Wayne Meissner
 * *
 * This file is part of jnr.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272166_1643050029.77/0/jnr-netdb-1-1-6-sources-jar/jnr/netdb/IANAServicesDB.java
1.1063 array-slice 1.1.0 1.1.0

1.1063.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1064 jsreport-scripts 2.6.0 2.6.0

1.1064.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007
This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the
facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:
0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1065 xmlhttprequest-ssl 1.5.3 1.5.3

1.1065.1 Available under license:
Copyright (c) 2010 passive.ly LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1066 tiger-types 1.4
1.1066.1 Available under license:
Open Source Used In DNAC Platform Arica 3756
Common Development and Distribution License (CDDL-1.0)
function myFunction() {
    var x;
    if (confirm("Disclaimer: While the OSI acknowledges these as potentially helpful resources for the community, it does not endorse any content, contributors or license interpretations from these websites. Any links to these resources across opensource.org are solely for navigational purposes. The OSI does not promote or exclusively favor any of the mentioned resources, but instead provides them as separate third-party resource to help inform your opinion. Any content from or links to these resources are separate from the OSI, exist for purely informational purposes and creates no attorney-client relationship between you, the OSI or the resources. If you have questions about how licenses apply to you or your organization, you should seek legal advice.") == true) {
        x = "<br><p>The following are other community resources that may be helpful:<br><br><a href=https://tldrlegal.com/license/common-development-and-distribution-license-%28cddl-1.0%29-explained style='font-weight: bold;'>Common Development and Distribution License (CDDL-1.0) on TLDRLegal<br><a href=http://www.gnu.org/licenses/license-list.en.html>GNU License List<br><a href=https://en.wikipedia.org/wiki/Comparison_of_free_and_open-source_software_licenses>Wikipedia License List<br><a href=http://oss-watch.ac.uk/apps/licdiff/>OSSWatch License Diff<br><a href=choosealicense.com>Choosealicense";
    } else {
        x = " ";
    }
    document.getElementById("demo").innerHTML = x;
}

//-->]]>
</script></div>

<b>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE</b> 
Version 1.0 (CDDL-1.0)</b> 
(<a href="cddl1.txt">text</a>)

<ul><li><p><b>1. Definitions.</b></p>
<li><p> <b>1.1. Contributor</b> means each individual or entity that creates or contributes to the creation of Modifications.</p></li><li><p> <b>1.2. Contributor Version</b> means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.</p></li><li><p> <b>1.3. Covered Software</b> means (a)
the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.</p>

1.4. Executable means the Covered Software in any form other than Source Code.</p>

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.</p>

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.</p>

1.7. License means this document.</p>

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.</p>

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.</p>

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the
common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</p>

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</p>

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof). The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the
combination of the Original Software with other software or devices.</p><li><p><b>2.2. Contributor Grant.</b></p><ul><li><p>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:</p><ul><li><p>(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and</p></li><li><p>(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).</p></li><li><p>(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.</p></li><li><p>(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.</p></li></ul></li></ul></li><li><p><b>3. Distribution Obligations.</b></p><ul><li><p><b>3.1. Availability of Source Code.</b></p></li></ul></li></ul>
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any
subsequent version of the License, You must distribute and
make the Covered Software available under the terms of the
version of the License under which You originally received
the Covered Software. Otherwise, You may also choose to
use, distribute or otherwise make the Covered Software
available under the terms of any subsequent version of the
License published by the license steward.</p>

4.3. Modified Versions.
When You are an Initial Developer and You want to
create a new license for Your Original Software, You may
create and use a modified version of this License if You:
(a) rename the license and remove any references to
the name of the license steward (except to note that the
license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ
from this License.</p>

5. DISCLAIMER OF WARRANTY.
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN
AS IS BASIS, WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS,
MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.
THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER
OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY
SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY
CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY
COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS
DISCLAIMER.</p>

6. TERMINATION.
6.1. This License and the rights granted
hereunder will terminate automatically if You fail to comply
with terms herein and fail to cure such breach within 30
days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination
of this License shall survive.
If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.
consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and
Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
1.1067 hbase-http 2.4.2

1.1067.1 Available under license:

Apache HBase - HTTP
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of MERCHANTABILITY, NON-INFRINGEMENT, TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.1068 @types/node 12.12.68 12.12.68

1.1068.1 Available under license:
MIT License

Copyright (c) Microsoft Corporation.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1069 amcharts3-dataloader 1.0.19 1.0.19

1.1069.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1070 core-js 2.5.7 2.5.7
1.1070.1 Available under license:
Copyright (c) 2014-2018 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1071 bcel 2.7.2
1.1071.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *     http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/* $Id: Version.src 1225426 2011-12-29 04:13:08Z mrglavas $ */
package org.apache.xalan;
/**
 * Administrative class to keep track of the version number of
 * the Xalan release.
 * <P>This class implements the upcoming standard of having
 * org.apache.project-name.Version.getVersion() be a standard way
 * to get version information. This class will replace the older
 * org.apache.xalan.processor.Version class.</P>
 * <P>See also: org/apache/xalan/res/XSLTInfo.properties for
 * information about the version of the XSLT spec we support.</P>
 * @xsl.usage general
 */
public class Version {

/**
 * Get the basic version string for the current Xalan release.
 * Version String formatted like
 * <CODE>"<B>Xalan</B> <B>Java</B> v.r[.dd| <B>D</B>nn"]</CODE>
 * Futurework: have this read version info from jar manifest.
 * @return String denoting our current version
 */
public static String getVersion() {
    return getProduct() + " " + getImplementationLanguage() + " "
            + getMajorVersionNum() + "." + getReleaseVersionNum() + "." +
            (getDevelopmentVersionNum() > 0) ? ("D" + getDevelopmentVersionNum()) : ("" + getMaintenanceVersionNum());
}

/**
 * Print the processor version to the command line.
 * @param argv command line arguments, unused.
 */
public static void main(String argv[]) {
    System.out.println(getVersion());
}

/**
 * Name of product: Xalan.
 */
public static String getProduct() {
    return "Xalan";
}
/**
 * Implementation Language: Java.
 */

public static String getImplementationLanguage()
{
    return "Java";
}

/**
 * Major version number.
 * Version number. This changes only when there is a significant, externally apparent enhancement from the previous release. 'n' represents the n'th version.
 * Clients should carefully consider the implications of new versions as external interfaces and behaviour may have changed.
 */

public static int getMajorVersionNum()
{
    return @version.VERSION@;
}

/**
 * Release Number.
 * Release number. This changes when:
 * - a new set of functionality is to be added, eg, implementation of a new W3C specification.
 * - API or behaviour change.
 * - its designated as a reference release.
 */

public static int getReleaseVersionNum()
{
    return @version.RELEASE@;
}

/**
 * Maintenance Drop Number.
 * Optional identifier used to designate maintenance drop applied to a specific release and contains fixes for defects reported. It maintains compatibility with the release and contains no API changes.
 * When missing, it designates the final and complete
*
development drop for a release.
*/
public static int getMaintenanceVersionNum()
{
return @version.MINOR@;
}
/**
* Development Drop Number.
* Optional identifier designates development drop of
*
a specific release. D01 is the first development drop
*
of a new release.
*
*
Development drops are works in progress towards a
*
compeleted, final release. A specific development drop
*
may not completely implement all aspects of a new
*
feature, which may take several development drops to
*
complete. At the point of the final drop for the
*
release, the D suffix will be omitted.
*
*
Each 'D' drops can contain functional enhancements as
*
well as defect fixes. 'D' drops may not be as stable as
*
the final releases.
*/
public static int getDevelopmentVersionNum()
{
try {
if ((new String("@version.DEVELOPER@")).length() == 0)
return 0;
else
return Integer.parseInt("@version.DEVELOPER@");
} catch (NumberFormatException nfe) {
return 0;
}
}
}
Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/Version.src
No license file was found, but licenses were detected in source scan.
/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

Open Source Used In DNAC Platform Arica 3787


You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* This is a special exception that is used to stop parsing when
* search for an element. For instance, when searching for xml:stylesheet
* PIs, it is used to stop the parse once the document element is found.
* @see StylesheetPIHandler
* @xsl.usage internal
*/

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xml/utils/StopParseException.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/compiler/XPathLexer.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file
package org.apache.xml.serializer;

/**
 * Administrative class to keep track of the version number of
 * the Serializer release.
 * <P>This class implements the upcoming standard of having
 * org.apache.project-name.Version.getVersion() be a standard way
 * to get version information.</P>
 * @xsl.usage general
 */
public final class Version 
{
    /**
     * Get the basic version string for the current Serializer.
     * Version String formatted like
     * <CODE>"<B>Serializer</B> <B>Java</B> v.r[.dd| <B>D</B>nn"]</CODE>
     *
     * Futurework: have this read version info from jar manifest.
     *
     * @return String denoting our current version
     */
    public static String getVersion() 
    { 
        return getProduct()+"+getImplementationLanguage()+"+
        +getMajorVersionNum()+"+"+getReleaseVersionNum()+"+"+
        +( (getDevelopmentVersionNum() > 0) ?
        ("D"+getDevelopmentVersionNum()) : (""+getMaintenanceVersionNum());
    } 
 */
* Print the processor version to the command line.
*
* @param argv command line arguments, unused.
*/
public static void main(String argv[])
{
    System.out.println(getVersion());
}

/**
* Name of product: Serializer.
 */
public static String getProduct()
{
    return "Serializer";
}

/**
* Implementation Language: Java.
*/
public static String getImplementationLanguage()
{
    return "Java";
}

/**
* Major version number.
* Version number. This changes only when there is a
* significant, externally apparent enhancement from
* the previous release. 'n' represents the n'th
* version.
* 
* Clients should carefully consider the implications
* of new versions as external interfaces and behaviour
* may have changed.
*/
public static int getMajorVersionNum()
{
    return @version.VERSION@;
}

/**
* Release Number.
* Release number. This changes when:
*   - a new set of functionality is to be added, eg,
*     implementation of a new W3C specification.
/* - API or behaviour change.
* - its designated as a reference release.
*/

public static int getReleaseVersionNum()
{
  return @version.RELEASE@;
}

/**
 * Maintenance Drop Number.
 * Optional identifier used to designate maintenance
 *   drop applied to a specific release and contains
 *   fixes for defects reported. It maintains compatibility
 *   with the release and contains no API changes.
 *   When missing, it designates the final and complete
 *   development drop for a release.
 */
public static int getMaintenanceVersionNum()
{
  return @version.MINOR@;
}

/**
 * Development Drop Number.
 * Optional identifier designates development drop of
 *   a specific release. D01 is the first development drop
 *   of a new release.
 *   Development drops are works in progress towards a
 *   completed, final release. A specific development drop
 *   may not completely implement all aspects of a new
 *   feature, which may take several development drops to
 *   complete. At the point of the final drop for the
 *   release, the D suffix will be omitted.
 *   Each 'D' drops can contain functional enhancements as
 *   well as defect fixes. 'D' drops may not be as stable as
 *   the final releases.
 */
public static int getDevelopmentVersionNum()
{
  try {
    if ((new String(@version.DEVELOPER@)).length() == 0)
      return 0;
    else
      return Integer.parseInt(@version.DEVELOPER@);
  } catch (NumberFormatException nfe) {
    return 0;
  }
Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/Version.src
No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Major version number.
 * Version number. This changes only when there is a
 * significant, externally apparent enhancement from
 * the previous release. 'n' represents the n'th
 * version.
 *
 * Clients should carefully consider the implications
 * of new versions as external interfaces and behaviour
 * may have changed.
 */

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/Version.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/XSLProcessorVersion.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/Version.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

// Attribution to: "Voytenko, Dimitry" <DVoytenko@SECTORBASE.COM>

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/ToHTMLStream.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

// Proprietary
/** The 'document-location()' id (Proprietary). */

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/compiler/FunctionTable.java

No license file was found, but licenses were detected in source scan.

# to you under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/Encodings.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/XMLEntities.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/output_text.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/output_xml.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/output_unknown.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xml/serializer/HTMLEntities.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xalan/res/XSLTInfo.properties
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xalan/xsltc/compiler/Makefile.inc

No license file was found, but licenses were detected in source scan.

/ *
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/
*/
$Id: xpath.cup 1225752 2011-12-30 04:12:46Z mrglavas $
*/
/
*/
* @author Jacek Ambroziak
* @author Santiago Pericas-Geertsen
package org.apache.xalan.xslt.compiler;

import java.util.Stack;
import java.util.Vector;
import java.io.StringReader;
import java_cup.runtime.*;
import org.apache.xml.dtm.DTM;
import org.apache.xalan.xslt.DOM;
import org.apache.xml.dtm.Axis;
import org.apache.xalan.xslt.runtime.Operators;
import org.apache.xalan.xslt.compiler.util.ErrorMsg;

parser code {:
   /**
    * Used by function calls with no args.
    */
   static public final Vector EmptyArgs = new Vector(0);

   /**
    * Reference to non-existing variable.
    */
   static public final VariableRef DummyVarRef = null;

   /**
    * Reference to the Parser class.
    */
   private Parser _parser;
   private XSLTC _xsltc;

   /**
    * String representation of the expression being parsed.
    */
   private String _expression;

   /**
    * Line number where this expression/pattern was declared.
    */
   private int _lineNumber = 0;

   /**
    * Reference to the symbol table.
    */
   public SymbolTable _symbolTable;
public XPathParser(Parser parser) {
    _parser = parser;
    _xsltc = parser.getXSLTC();
    _symbolTable = parser.getSymbolTable();
}

public int getLineNumber() {
    return _lineNumber;
}

public QName getQNameIgnoreDefaultNs(String name) {
    return _parser.getQNameIgnoreDefaultNs(name);
}

public QName getQName(String namespace, String prefix, String localname) {
    return _parser.getQName(namespace, prefix, localname);
}

public void setMultiDocument(boolean flag) {
    _xsltc.setMultiDocument(flag);
}

public void setCallsNodeset(boolean flag) {
    _xsltc.setCallsNodeset(flag);
}

public void setHasIdCall(boolean flag) {
    _xsltc.setHasIdCall(flag);
}

/**
 * This method is similar to findNodeType(int, Object) except that it
 * creates a StepPattern instead of just returning a node type. It also
 * differs in the way it handles "*[uri]:*" and "*[uri]:@*". The last two
 * patterns are expanded as "*[namespace-uri() = 'uri']" and
 * "[@*[namespace-uri() = 'uri']", respectively. This expansion considerably
 * simplifies the grouping of patterns in the Mode class. For this
 * expansion to be correct, the priority of the pattern/template must be
 * set to -0.25 (when no other predicates are present).
 */

public StepPattern createStepPattern(int axis, Object test, Vector predicates) {
    int nodeType;

    if (test == null) {  // "*
        nodeType = (axis == Axis.ATTRIBUTE) ? NodeTest.ATTRIBUTE :
                    (axis == Axis.NAMESPACE) ? -1 : NodeTest.ELEMENT:


return new StepPattern(axis, nodeType, predicates);
}
else if (test instanceof Integer) {
    nodeType = ((Integer) test).intValue();
}
else {
    QName name = (QName)test;
    boolean setPriority = false;

    if (axis == Axis.NAMESPACE) {
        nodeType = (name.toString().equals("*")) ? -1
            : _xsltc.registerNamespacePrefix(name);
    } else {
        final String uri = name.getNamespace();
        final String local = name.getLocalPart();
        final QName namespace_uri =
            _parser.getQNameIgnoreDefaultNs("namespace-uri");

        // Expand {uri}:* to *[namespace-uri() = 'uri'] - same for @*
        if (uri != null && (local.equals("*") || local.equals("@*"))) {
            if (predicates == null) {
                predicates = new Vector(2);
            }
            predicates.add( /* Constructor */
                new Predicate("namespace-uri", uri));
        }

        // Priority is set by hand if no other predicates exist
        setPriority = (predicates.size() == 0);

        predicates.add("namespace-uri", uri);
        predicates.add(new Predicate("namespace-uri", namespace_uri));
        predicates.add(new Predicate("local", uri));
    }

    if (local.equals("*")) {
        nodeType = (axis == Axis.ATTRIBUTE) ? NodeTest.ATTRIBUTE
            : NodeTest.ELEMENT;
    } else if (local.equals("@*")) {
        nodeType = NodeTest.ATTRIBUTE;
    } else {
        nodeType = (axis == Axis.ATTRIBUTE) ? _xsltc.registerAttribute(name)
            : _xsltc.registerElement(name);
final StepPattern result = new StepPattern(axis, nodeType, predicates);

// Set priority for case prefix:* and prefix:@* (no predicates)
if (setPriority) {
    result.setPriority(-0.25);
}

return result;

}

public int findNodeType(int axis, Object test) {
if (test == null) { // *
    return (axis == Axis.ATTRIBUTE) ?
            NodeTest.ATTRIBUTE :
            (axis == Axis.NAMESPACE) ? -1 :
            NodeTest.ELEMENT;
}
else if (test instanceof Integer) {
    return ((Integer)test).intValue();
}
else {
    QName name = (QName)test;

    if (axis == Axis.NAMESPACE) {
        return (name.toString().equals("*")) ? -1 :
            _xsltc.registerNamespacePrefix(name);
    }
    else if (name.getNamespace() == null) {
        final String local = name.getLocalPart();

        if (local.equals("*")) {
            return (axis == Axis.ATTRIBUTE) ? NodeTest.ATTRIBUTE :
                        NodeTest.ELEMENT;
        }
        else if (local.equals("@*")) {
            return NodeTest.ATTRIBUTE;
        }
    }

    if (name.getNamespace() == null) {
        final String local = name.getLocalPart();

        if (local.equals("*")) {
            return (axis == Axis.ATTRIBUTE) ? _xsltc.registerAttribute(name) :
                        _xsltc.registerElement(name);
        }
    }

    return (axis == Axis.ATTRIBUTE) ? _xsltc.registerAttribute(name) :
                        _xsltc.registerElement(name);
}
public Symbol parse(String expression, int lineNumber) throws Exception {
    try {
        _expression = expression;
        _lineNumber = lineNumber;
        return super.parse();
    }
    catch (IllegalCharException e) {
        ErrorMsg err = new ErrorMsg(ErrorMsg.ILLEGAL_CHAR_ERR,
                                     lineNumber, e.getMessage());
        _parser.reportError(Constants.FATAL, err);
    }
    return null;
}

/**
 * Lookup a variable or parameter in the symbol table given its name.
 * @param name Name of the symbol being looked up.
 */
final SyntaxTreeNode lookupName(QName name) {
    // Is it a local var or param ?
    final SyntaxTreeNode result = _parser.lookupVariable(name);
    if (result != null)
        return(result);
    else
        return(_symbolTable.lookupName(name));
}

/**
 * Add an error.
 * @param error Error message.
 */
public final void addError(ErrorMsg error) {
    _parser.reportError(Constants.ERROR, error);
}

public void report_error(String message, Object info) {

    final ErrorMsg err = new ErrorMsg(ErrorMsg.SYNTAX_ERR, _lineNumber,
                                      _expression);
    _parser.reportError(Constants.FATAL, err);
}

/*
 * Parse the expression passed to the current scanner. If this
 * expression contains references to local variables and it will be
 * compiled in an external module (not in the main class) request
 * the current template to create a new variable stack frame.
 *
 * @param lineNumber Line where the current expression is defined.
 * @param external   Set to <tt>true</tt> if this expression is
 *                   compiled in a separate module.
 */

public Symbol parse(String expression, int lineNumber) throws Exception {
    try {
        _expression = expression;
        _lineNumber = lineNumber;
        return super.parse();
    }
    catch (IllegalCharException e) {
        ErrorMsg err = new ErrorMsg(ErrorMsg.ILLEGAL_CHAR_ERR,
                                     lineNumber, e.getMessage());
        _parser.reportError(Constants.FATAL, err);
    }
    return null;
}
public void report_fatal_error(String message, Object info) {
    // empty
}

public RelativeLocationPath insertStep(Step step, RelativeLocationPath rlp) {
    if (rlp instanceof Step) {
        return new ParentLocationPath(step, (Step) rlp);
    }
    else if (rlp instanceof ParentLocationPath) {
        final ParentLocationPath plp = (ParentLocationPath) rlp;
        final RelativeLocationPath newrlp = insertStep(step, plp.getPath());
        return new ParentLocationPath(newrlp, plp.getStep());
    }
    else {
        addError(new ErrorMsg(ErrorMsg.INTERNAL_ERR, "XPathParser.insertStep");
        return rlp;
    }
}

/**
 * Returns true if the axis applies to elements only. The axes
 * child, attribute, namespace, descendant result in non-empty
 * nodesets only if the context node is of type element.
 */
public boolean isElementAxis(int axis) {
    return (axis == Axis.CHILD || axis == Axis.ATTRIBUTE ||
            axis == Axis.NAMESPACE || axis == Axis.DESCENDANT);
}

terminal SLASH, DOT, LBRACK, RBRACK, VBAR, LPAREN, RPAREN, STAR, COMMA;
terminal DOLLAR, ATSIGN;
terminal DDOT, DCOLON, DSLASH;
terminal EQ, NE;
terminal LT, GT, LE, GE;
terminal PLUS, MINUS, DIV, MOD, MULT;
terminal String Literal;
terminal String QNAME;
terminal ID, KEY, TEXT, NODE, OR, AND, COMMENT, PI, PIPARAM, PRECEDINGSIBLING;
terminal SELF, PARENT, CHILD, ATTRIBUTE, ANCESTOR, ANCESTORORSELF, DESCENDANT;
terminal DESCENDANTORSELF, FOLLOWING, FOLLOWINGSIBLING, NAMESPACE, PRECEDING;
terminal Double REAL;
terminal Long INT;
terminal PATTERN, EXPRESSION;

non terminal SyntaxTreeNode TopLevel;
non terminal Expression Expr, Argument, LocationPath;
non terminal Expression Predicate, FilterExpr, Step;
non terminal Expression OrExpr, AndExpr, EqualityExpr;
non terminal Expression RelationalExpr, AdditiveExpr;
non terminal Expression MultiplicativeExpr, UnaryExpr;
non terminal Expression VariableReference, FunctionCall;
non terminal Expression PrimaryExpr, UnionExpr, PathExpr, AbbreviatedStep;
non terminal Expression RelativeLocationPath, AbbreviatedRelativeLocationPath;
non terminal Expression AbsoluteLocationPath, AbbreviatedAbsoluteLocationPath;

non terminal Object NodeTest, NameTest;

non terminal IdKeyPattern IdKeyPattern;
non terminal Pattern Pattern;
non terminal Pattern LocationPathPattern;
non terminal StepPattern ProcessingInstructionPattern;
non terminal RelativePathPattern RelativePathPattern;
non terminal StepPattern StepPattern;
non terminal Object NodeTestPattern, NameTestPattern;

non terminal Vector Predicates, NonemptyArgumentList;
non terminal QName QName, FunctionName, VariableName;
non terminal Integer AxisName, AxisSpecifier;
non terminal Integer ChildOrAttributeAxisSpecifier;

precedence left VBAR;
precedence left OR;
precedence left AND;
precedence nonassoc EQ, NE;
precedence left LT, GT, LE, GE;

precedence left PLUS, MINUS;
precedence left DIV, MOD, MULT;
precedence left DOLLAR;
precedence left ATSIGN;
precedence right DCOLON;

start with TopLevel;

TopLevel ::= PATTERN Pattern:pattern
            [: RESULT = pattern; :]

            | EXPRESSION Expr:expr
            [: RESULT = expr; :];

    /* --------------------------- Patterns ----------------------------------- */

Pattern ::= LocationPathPattern:lpp
LocationPathPattern ::= SLASH
  { : RESULT = new AbsolutePathPattern(null); : }

SLASH RelativePathPattern::rpp
  { : RESULT = new AbsolutePathPattern(rpp); : }

IdKeyPattern::ikp
  { : RESULT = ikp; : }

IdKeyPattern::ikp SLASH RelativePathPattern::rpp
  { : RESULT = new ParentPattern(ikp, rpp); : }

IdKeyPattern::ikp DSLASH RelativePathPattern::rpp
  { : RESULT = new AncestorPattern(ikp, rpp); : }

DSLASH RelativePathPattern::rpp
  { : RESULT = new AncestorPattern(rpp); : }

RelativePathPattern::rpp
  { : RESULT = rpp; : }

IdKeyPattern ::= ID LPAREN Literal:l RPAREN
  { : RESULT = new IdPattern(l);
    parser.setHasIdCall(true);
  : }

KEY LPAREN Literal:l1 COMMA Literal:l2 RPAREN
  { : RESULT = new KeyPattern(l1, l2); : }

ProcessingInstructionPattern ::= PIPARAM LPAREN Literal:l RPAREN
  { : RESULT = new ProcessingInstructionPattern(l); : }

RelativePathPattern ::= StepPattern::sp
  { : RESULT = sp; : }

StepPattern::sp SLASH RelativePathPattern::rpp
  { : RESULT = new ParentPattern(sp, rpp); : }

StepPattern::sp DSLASH RelativePathPattern::rpp
  { : RESULT = new AncestorPattern(sp, rpp); : }

StepPattern ::= NodeTestPattern::nt
  { :
RESULT = parser.createStepPattern(Axis.CHILD, nt, null);
    :}

| NodeTestPattern:nt Predicates:pp
| :
RESULT = parser.createStepPattern(Axis.CHILD, nt, pp);
    :}

| ProcessingInstructionPattern:pip
| :
RESULT = pip; }

| ProcessingInstructionPattern:pip Predicates:pp
| : RESULT = (ProcessingInstructionPattern)pip.setPredicates(pp); :

| ChildOrAttributeAxisSpecifier:axis NodeTestPattern:nt
| :
RESULT = parser.createStepPattern(axis.intValue(), nt, null);
    :

| ChildOrAttributeAxisSpecifier:axis
NodeTestPattern:nt Predicates:pp
| :
RESULT = parser.createStepPattern(axis.intValue(), nt, pp);
    :

| ChildOrAttributeAxisSpecifier:axis ProcessingInstructionPattern:pip
| :
RESULT = pip; // TODO: report error if axis is attribute
    :

| ChildOrAttributeAxisSpecifier:axis ProcessingInstructionPattern:pip
Predicates:pp
| :
// TODO: report error if axis is attribute
RESULT = (ProcessingInstructionPattern)pip.setPredicates(pp); :

NodeTestPattern ::= NameTestPattern:nt
| :
RESULT = nt; }

| NODE
| :
RESULT = new Integer(NodeTest.ANODE); :

| TEXT
| :
RESULT = new Integer(NodeTest(TEXT); :

| COMMENT
| :
RESULT = new Integer(NodeTest.COMMENT); :}
NameTestPattern ::= STAR
{: RESULT = null; :}

| QName:qn
{: RESULT = qn; :}

ChildOrAttributeAxisSpecifier ::= ATSIGN
{: RESULT = new Integer(Axis.ATTRIBUTE); :}

| CHILD DCOLON
{: RESULT = new Integer(Axis.CHILD); :}

| ATTRIBUTE DCOLON
{: RESULT = new Integer(Axis.ATTRIBUTE); :};

Predicates ::= Predicate:p
{: Vector temp = new Vector(); temp.addElement(p); RESULT = temp; :}

| Predicate:p Predicates:pp
{: pp.insertElementAt(p, 0); RESULT = pp; :};

Predicate ::= LBRACK Expr:e RBRACK
{: RESULT = new Predicate(e); :};

/* --------------------------- Expressions --------------------------------- */

Expr ::= OrExpr:ex
{: RESULT = ex; :}

OrExpr ::= AndExpr:ae
{: RESULT = ae; :}

| OrExpr:oe OR AndExpr:ae
{: RESULT = new LogicalExpr(LogicalExpr.OR, oe, ae); :};

AndExpr ::= EqualityExpr:e
{: RESULT = e; :}
[AndExpr:ae AND EqualityExpr:ee
  { RESULT = new LogicalExpr(LogicalExpr.AND, ae, ee); }]

EqualityExpr ::= RelationalExpr:re
  { RESULT = re; }

| EqualityExpr:ee EQ RelationalExpr:re
  { RESULT = new EqualityExpr(Operators.EQ, ee, re); }

| EqualityExpr:ee NE RelationalExpr:re
  { RESULT = new EqualityExpr(Operators.NE, ee, re); }

RelationalExpr ::= AdditiveExpr:ae
  { RESULT = ae; }

| RelationalExpr:re LT AdditiveExpr:ae
  { RESULT = new RelationalExpr(Operators.LT, re, ae); }

| RelationalExpr:re GT AdditiveExpr:ae
  { RESULT = new RelationalExpr(Operators.GT, re, ae); }

| RelationalExpr:re LE AdditiveExpr:ae
  { RESULT = new RelationalExpr(Operators.LE, re, ae); }

| RelationalExpr:re GE AdditiveExpr:ae
  { RESULT = new RelationalExpr(Operators.GE, re, ae); }

AdditiveExpr ::= MultiplicativeExpr:me
  { RESULT = me; }

| AdditiveExpr:ae PLUS MultiplicativeExpr:me
  { RESULT = new BinOpExpr(BinOpExpr.PLUS, ae, me); }

| AdditiveExpr:ae MINUS MultiplicativeExpr:me
  { RESULT = new BinOpExpr(BinOpExpr.MINUS, ae, me); }

MultiplicativeExpr ::= UnaryExpr:ue
  { RESULT = ue; }

| MultiplicativeExpr:me MULT UnaryExpr:ue
  { RESULT = new BinOpExpr(BinOpExpr.TIMES, me, ue); }

| MultiplicativeExpr:me DIV UnaryExpr:ue
  { RESULT = new BinOpExpr(BinOpExpr.DIV, me, ue); }

| MultiplicativeExpr:me MOD UnaryExpr:ue
  { RESULT = new BinOpExpr(BinOpExpr.MOD, me, ue); }
UnaryExpr ::= UnionExpr:ue
   {: RESULT = ue; :}

   | MINUS UnaryExpr:ue
     {: RESULT = new UnaryOpExpr(ue); :};

UnionExpr ::= PathExpr:pe
   {: RESULT = pe; :}

   | PathExpr:pe VBAR UnionExpr:rest
     {: RESULT = new UnionPathExpr(pe, rest); :};

PathExpr ::= LocationPath:lp
   {: RESULT = lp; :}

   | FilterExpr:fexp
     {: RESULT = fexp; :}

   | FilterExpr:fexp SLASH RelativeLocationPath:rlp
     {: RESULT = new FilterParentPath(fexp, rlp); :}

   | FilterExpr:fexp DSLASH RelativeLocationPath:rlp
     {:
       // Expand ‘/’ into ’/descendant-or-self::node()’ or
       // into /descendant-or-self::*/
       //
       int nodeType = DOM.NO_TYPE;
       if (rlp instanceof Step &&
           parser.isElementAxis(((Step) rlp).getAxis()))
       {
         nodeType = DTM.ELEMENT_NODE;
       }
       final Step step = new Step(Axis.DESCENDANTORSELF, nodeType, null);
       FilterParentPath fpp = new FilterParentPath(fexp, step);
       fpp = new FilterParentPath(fpp, rlp);
       if (fexp instanceof KeyCall == false) {
         fpp.setDescendantAxis();
       }
     }
     RESULT = fpp;
     :};

LocationPath ::= RelativeLocationPath:rlp
   {: RESULT = rlp; :}

   | AbsoluteLocationPath:alp
     {: RESULT = alp; :};
RelativeLocationPath ::= Step:step
{: RESULT = step; :}

| RelativeLocationPath:rlp SLASH Step:step
{:}
if (rlp instanceof Step && ((Step) rlp).isAbbreviatedDot()) {
    RESULT = step;       // Remove './' from the middle
}
else if (((Step) step).isAbbreviatedDot()) {
    RESULT = rlp;        // Remove './' from the end
}
else {
    RESULT =
    new ParentLocationPath((RelativeLocationPath) rlp, step);
}

| AbbreviatedRelativeLocationPath:arlp
{: RESULT = arlp; :};

AbsoluteLocationPath ::= SLASH
{: RESULT = new AbsoluteLocationPath(); :}

| SLASH RelativeLocationPath:rlp
{: RESULT = new AbsoluteLocationPath(rlp); :}

| AbbreviatedAbsoluteLocationPath:aalp
{: RESULT = aalp; :};

AbbreviatedRelativeLocationPath ::= RelativeLocationPath:rlp DSLASH Step:step
{:}
final Step right = (Step)step;
final int axis = right.getAxis();
final int type = right.getNodeType();
final Vector predicates = right.getPredicates();

if ((axis == Axis.CHILD) && (type != NodeTest.ATTRIBUTE)) {
// Compress './child:E' into 'descendant::E' - if possible
if (predicates == null) {
    right.setAxis(Axis.DESCENDANT);
    if (rlp instanceof Step && ((Step)rlp).isAbbreviatedDot()) {
        RESULT = right;
    }
    else {
        // Expand 'rlp/child::E' into 'rlp/descendant::E'
        RelativeLocationPath left = (RelativeLocationPath)rlp;
        RESULT = new ParentLocationPath(left, right);
    }
}
void expandSteps()
{
    // Expand '//step' -> 'descendant-or-self::*/*/step'
    if (rlp instanceof Step && ((Step)rlp).isAbbreviatedDot()) {
        Step left = new Step(Axis.DESCENDANTORSELF,
                DTM.ELEMENT_NODE, null);
        RESULT = new ParentLocationPath(left, right);
    }
    else {
        // Expand 'rlp//step' -> 'rlp/descendant-or-self::*/*/step'
        RelativeLocationPath left = (RelativeLocationPath)rlp;
        Step mid = new Step(Axis.DESCENDANTORSELF,
                DTM.ELEMENT_NODE, null);
        ParentLocationPath ppl = new ParentLocationPath(mid, right);
        RESULT = new ParentLocationPath(left, ppl);
    }
}

void expandAttributes()
{
    RelativeLocationPath left = (RelativeLocationPath)rlp;
    Step middle = new Step(Axis.DESCENDANTORSELF,
            DOM.NO_TYPE, null);
    ParentLocationPath ppl = new ParentLocationPath(middle, right);
    RESULT = new ParentLocationPath(left, ppl);
}

void expandNodes()
{
    RelativeLocationPath left = (RelativeLocationPath)rlp;
    Step middle = new Step(Axis.DESCENDANTORSELF,
            DOM.NO_TYPE, null);
    ParentLocationPath ppl = new ParentLocationPath(middle, right);
    RESULT = new ParentLocationPath(left, ppl);
}

// Expand '//' into '/descendant-or-self::node()/' or
// into /descendant-or-self::*/
int nodeType = DOM.NO_TYPE;
if (rlp instanceof Step &&
        parser.isElementAxis(((Step) rlp).getAxis()))
{
    nodeType = DTM.ELEMENT_NODE;
}

AbbreviatedAbsoluteLocationPath ::= DSLASH RelativeLocationPath:rlp

;
final Step step = new Step(Axis.DESCENDANTORSELF, nodeType, null);
RESULT = new AbsoluteLocationPath(parser.insertStep(step, (RelativeLocationPath) rlp));
);

Step ::= NodeTest:ntest
{: if (ntest instanceof Step) {
   RESULT = (Step)ntest;
}
else {
RESULT = new Step(Axis.CHILD,
   parser.findNodeType(Axis.CHILD, ntest),
   null);
}
:}

| NodeTest:ntest Predicates:pp
{: if (ntest instanceof Step) {
   Step step = (Step)ntest;
   step.addPredicates(pp);
   RESULT = (Step)ntest;
}
else {
RESULT = new Step(Axis.CHILD,
   parser.findNodeType(Axis.CHILD, ntest),
   pp);
}
:}

| AxisSpecifier:axis NodeTest:ntest Predicates:pp
{: RESULT = new Step(axis.intValue(),
   parser.findNodeType(axis.intValue(), ntest),
   pp);
:}

| AxisSpecifier:axis NodeTest:ntest
{: RESULT = new Step(axis.intValue(),
   parser.findNodeType(axis.intValue(), ntest),
   null);
:}

| AbbreviatedStep:abbrev
{: RESULT = abbrev; :};

AxisSpecifier ::= AxisName:an DCOLON
{: RESULT = an; :}
| ATSIGN      |
|            |
| {} RESULT = new Integer(Axis.ATTRIBUTE); | |

AxisName ::= ANCESTOR
{}
RESULT = new Integer(Axis.ANCESTOR); |

| ANCESTORORSELF |
|                |
| {} RESULT = new Integer(Axis.ANCESTORORSELF); |

| ATTRIBUTE |
|          |
| {} RESULT = new Integer(Axis.ATTRIBUTE); |

| CHILD |
|       |
| {} RESULT = new Integer(Axis.CHILD); |

| DESCENDANT |
|           |
| {} RESULT = new Integer(Axis.DESCENDANT); |

| DESCENDANTORSELF |
|                  |
| {} RESULT = new Integer(Axis.DESCENDANTORSELF); |

| FOLLOWING |
|          |
| {} RESULT = new Integer(Axis.FOLLOWING); |

| FOLLOWINGSIBLING |
|                  |
| {} RESULT = new Integer(Axis.FOLLOWINGSIBLING); |

| NAMESPACE |
|          |
| {} RESULT = new Integer(Axis.NAMESPACE); |

| PARENT |
|        |
| {} RESULT = new Integer(Axis.PARENT); |

| PRECEDING |
|          |
| {} RESULT = new Integer(Axis.PRECEDING); |

| PRECEDINGSIBLING |
|                 |
| {} RESULT = new Integer(Axis.PRECEDINGSIBLING); |

| SELF |
|      |
| {} RESULT = new Integer(Axis.SELF); |

AbbreviatedStep ::= DOT
{}
RESULT = new Step(Axis.SELF, NodeTest.ANODE, null); |

| DDOT |
|      |
| {} RESULT = new Step(Axis.PARENT, NodeTest.ANODE, null); |
FilterExpr ::= PrimaryExpr:primary
  { : RESULT = primary; : }

  | PrimaryExpr:primary Predicates:pp
  { : RESULT = new FilterExpr(primary, pp); : }

PrimaryExpr ::= VariableReference:vr
  { : RESULT = vr; : }

  | LPAREN Expr:ex RPAREN
  { : RESULT = ex; : }

  | Literal:string
  { :
  /*
  * If the string appears to have the syntax of a QName, store
  * namespace info in the literal expression. This is used for
  * element-available and function-available functions, among
  * others. Also, the default namespace must be ignored.
  */
  String namespace = null;
  final int index = string.lastIndexOf(':');

  if (index > 0) {
    final String prefix = string.substring(0, index);
    namespace = parser._symbolTable.lookupNamespace(prefix);
  }
  RESULT = (namespace == null) ? new LiteralExpr(string)
    : new LiteralExpr(string, namespace);
  :

  | INT:num
  { :
    long value = num.longValue();
    if (value < Integer.MIN_VALUE || value > Integer.MAX_VALUE) {
      RESULT = new RealExpr(value);
    }
    else {
      if (num.doubleValue() == -0)
        RESULT = new RealExpr(num.doubleValue());
      else if (num.intValue() == 0)
        RESULT = new IntExpr(num.intValue());
      else if (num.doubleValue() == 0.0)
        RESULT = new RealExpr(num.doubleValue());
      else
        RESULT = new IntExpr(num.intValue());
    }
  }
VariableReference ::= DOLLAR VariableName:varName

if (node != null) {
  if (node instanceof Variable) {
    RESULT = new VariableRef((Variable)node);
  }
  else if (node instanceof Param) {
    RESULT = new ParameterRef((Param)node);
  }
  else {
    RESULT = new UnresolvedRef(varName);
  }
}

if (node == null) {
  RESULT = new UnresolvedRef(varName);
}

FunctionCall ::= FunctionName:fname LPAREN RPAREN

if (parser.getQNameIgnoreDefaultNs("current").equals(fname)) {
  RESULT = new CurrentCall(fname);
}
else if (parser.getQNameIgnoreDefaultNs("number").equals(fname)) {
  RESULT = new NumberCall(fname, parser.EmptyArgs);
}
else if (parser.getQNameIgnoreDefaultNs("string").equals(fname)) {
  RESULT = new StringCall(fname, parser.EmptyArgs);
}
else if (parser.getQNameIgnoreDefaultNs("concat").equals(fname)) {
  RESULT = new ConcatCall(fname, parser.EmptyArgs);
}
else if (parser.getQNameIgnoreDefaultNs("true").equals(fname)) {
  RESULT = new BooleanExpr(true);
}
else if (parser.getQNameIgnoreDefaultNs("false").equals(fname)) {
    RESULT = new BooleanExpr(false);
}
else if (parser.getQNameIgnoreDefaultNs("name").equals(fname)) {
    RESULT = new NameCall(fname);
}
else if (parser.getQNameIgnoreDefaultNs("generate-id").equals(fname)) {
    RESULT = new GenerateIdCall(fname, parser.EmptyArgs);
}
else if (parser.getQNameIgnoreDefaultNs("string-length").equals(fname)) {
    RESULT = new StringLengthCall(fname, parser.EmptyArgs);
}
else if (parser.getQNameIgnoreDefaultNs("position").equals(fname)) {
    RESULT = new PositionCall(fname);
}
else if (parser.getQNameIgnoreDefaultNs("last").equals(fname)) {
    RESULT = new LastCall(fname);
}
else if (parser.getQNameIgnoreDefaultNs("local-name").equals(fname)) {
    RESULT = new LocalNameCall(fname);
}
else if (parser.getQNameIgnoreDefaultNs("namespace-uri").equals(fname)) {
    RESULT = new NamespaceUriCall(fname);
}
else {
    RESULT = new FunctionCall(fname, parser.EmptyArgs);
}
:
|
| FunctionName:fname LPAREN NonemptyArgumentList:argl RPAREN
|
:  |
| if (parser.getQNameIgnoreDefaultNs("concat").equals(fname)) {
|   RESULT = new ConcatCall(fname, argl);
|
| else if (parser.getQNameIgnoreDefaultNs("number").equals(fname)) {
|   RESULT = new NumberCall(fname, argl);
|
| else if (parser.getQNameIgnoreDefaultNs("document").equals(fname)) {
|   parser.setMultiDocument(true);
|   RESULT = new DocumentCall(fname, argl);
|
| else if (parser.getQNameIgnoreDefaultNs("string").equals(fname)) {
|   RESULT = new StringCall(fname, argl);
|
| else if (parser.getQNameIgnoreDefaultNs("boolean").equals(fname)) {
|   RESULT = new BooleanCall(fname, argl);
|}
else if (parser.getQNameIgnoreDefaultNs("name").equals(fname)) {
    RESULT = new NameCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("generate-id").equals(fname)) {
    RESULT = new GenerateIdCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("not").equals(fname)) {
    RESULT = new NotCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("format-number").equals(fname)) {
    RESULT = new FormatNumberCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("unparsed-entity-uri").equals(fname)) {
    RESULT = new UnparsedEntityUriCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("key").equals(fname)) {
    RESULT = new KeyCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("id").equals(fname)) {
    RESULT = new KeyCall(fname, argl);
    parser.setHasIdCall(true);
}
else if (parser.getQNameIgnoreDefaultNs("ceiling").equals(fname)) {
    RESULT = new CeilingCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("round").equals(fname)) {
    RESULT = new RoundCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("floor").equals(fname)) {
    RESULT = new FloorCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("contains").equals(fname)) {
    RESULT = new ContainsCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("string-length").equals(fname)) {
    RESULT = new StringLengthCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("starts-with").equals(fname)) {
    RESULT = new StartsWithCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("function-available").equals(fname)) {
    RESULT = new FunctionAvailableCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("element-available").equals(fname)) {
    RESULT = new ElementAvailableCall(fname, argl);
}
else if (parser.getQNameIgnoreDefaultNs("local-name").equals(fname)) {
    RESULT = new LocalNameCall(fname, argl);
else if (parser.getQNameIgnoreDefaultNs("lang").equals(fname)) {
    RESULT = new LangCall(fname, argl);
}

else if (parser.getQNameIgnoreDefaultNs("namespace-uri").equals(fname)) {
    RESULT = new NamespaceUriCall(fname, argl);
}

else if (parser.getQName( Constants.TRANSLET_URI, "xsltc", "cast").equals(fname)) {
    RESULT = new CastCall(fname, argl);
}

// Special case for extension function nodeset()
else if (fname.getLocalPart().equals("nodeset") || fname.getLocalPart().equals("node-set")) {
    parser.setCallsNodeset(true);  // implies MultiDOM
    RESULT = new FunctionCall(fname, argl);
}
else {
    RESULT = new FunctionCall(fname, argl);
}
);

NonemptyArgumentList ::= Argument:arg
{: argl.insertElementAt(arg, 0); RESULT = argl; :};

FunctionName ::= QName:fname
{: RESULT = fname; :};

VariableName ::= QName:vname
{: RESULT = vname; :};

Argument ::= Expr:ex
{: RESULT = ex; :};

NodeTest ::= NameTest:nt
{: RESULT = nt; :}

| NODE
{: RESULT = new Integer(NodeTest.ANODE); :}
TEXT
{: RESULT = new Integer(NodeTest.TEXT); :}

COMMENT
{: RESULT = new Integer(NodeTest.COMMENT); :}

PI
{: RESULT = new Integer(NodeTest.PI); :}

NameTest ::= STAR
{: RESULT = null; :}

| QName:qn
{: RESULT = qn; :};

QName ::= QName:qname
{: RESULT = parser.getQNameIgnoreDefaultNs(qname); :}

| DIV
{: RESULT = parser.getQNameIgnoreDefaultNs("div"); :}

| MOD
{: RESULT = parser.getQNameIgnoreDefaultNs("mod"); :}

| KEY
{: RESULT = parser.getQNameIgnoreDefaultNs("key"); :}

| ANCESTOR
{: RESULT = parser.getQNameIgnoreDefaultNs("child"); :}

| ANCESTORORSELF
{: RESULT = parser.getQNameIgnoreDefaultNs("ancestor-or-self"); :}

| ATTRIBUTE
{: RESULT = parser.getQNameIgnoreDefaultNs("attribute"); :}
CHILD
{: RESULT = parser.getQNameIgnoreDefaultNs("child"); :}

DESCENDANT
{: RESULT = parser.getQNameIgnoreDefaultNs("descendant"); :}

DESCENDANTORSELF
{: RESULT = parser.getQNameIgnoreDefaultNs("descendant-or-self"); :}

FOLLOWING
{: RESULT = parser.getQNameIgnoreDefaultNs("following"); :}

FOLLOWINGSIBLING
{: RESULT = parser.getQNameIgnoreDefaultNs("following-sibling"); :}

NAMESPACE
{: RESULT = parser.getQNameIgnoreDefaultNs("namespace"); :}

PARENT
{: RESULT = parser.getQNameIgnoreDefaultNs("parent"); :}

PRECEDING
{: RESULT = parser.getQNameIgnoreDefaultNs("preceeding"); :}

PRECEDINGSIBLING
{: RESULT = parser.getQNameIgnoreDefaultNs("preceeding-sibling"); :}

SELF
{: RESULT = parser.getQNameIgnoreDefaultNs("self"); :}

ID
{: RESULT = parser.getQNameIgnoreDefaultNs("id"); :}

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/xpath.cup
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*/
/*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
// is found, then throw a special exception in order to terminate

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/TransformerFactoryImpl.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (c) 2002 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University). All
* Rights Reserved. This program is distributed under the W3C's Software
* Intellectual Property License. This program is distributed in the
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
* PURPOSE.
* See W3C License http://www.w3.org/Consortium/Legal/ for more details.
*/

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/domapi/XPathStylesheetDOM3Exception.java
No license file was found, but licenses were detected in source scan.

/
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * Execute the proprietary document-location() function, which returns
 * a node set of documents.
 * @xsl.usage advanced
 */

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/functions/FuncDoclocation.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package org.apache.xalan.xsltc.compiler;

import java_cup.runtime.Symbol;
%%%
cup
%unicode
%class XPathLexer
%yyeof
%

int last;

void initialize() {
    last = -1;
}

static boolean isWhitespace(int c) {
    return (c == ' ' || c == '	' || c == '' || c == '
' || c == '');
}

/**
* If symbol is not followed by '::' or '(', then treat it as a
* name instead of an axis or function (Jira-1912).
*/
Symbol disambiguateAxisOrFunction(int ss) throws Exception {
    // Peek in the input buffer without changing the internal state
    int index = yy_buffer_index;

    // Skip whitespace
    while (index < yy_buffer_read && isWhitespace(yy_buffer[index])) {
        index++;
    }

    // If end of buffer, can't disambiguate :
    if (index >= yy_buffer_read) {
        // Can't disambiguate, so return as symbol
        return new Symbol(ss);
    }

    // Return symbol if next token is '::' or '('
    return (yy_buffer[index] == ':' && yy_buffer[index+1] == ':') ||
           yy_buffer[index] == '(') ?
      newSymbol(ss) : newSymbol(sym.QNAME, yytext());
}

/**
* If symbol is first token or if it follows any of the operators
* listed in http://www.w3.org/TR/xpath#exprlex then treat as a
* name instead of a keyword (Jira-1912).
*/
Symbol disambiguateOperator(int ss) throws Exception {
    switch (last) {
    case -1:    // first token
    case sym.ATSIGN:
    case sym.DCOLON:
    case sym.LPAREN:
    case sym.LBRACK:
    case sym.COMMA:
case sym.AND:
case sym.OR:
case sym.MOD:
case sym.DIV:
case sym.MULT:
case sym.SLASH:
case sym.DSLSH:
case sym.VBAR:
case sym.PLUS:
case sym.MINUS:
case sym.EQ:
case sym.NE:
case sym.LT:
case sym.LE:
case sym.GT:
case sym.GE:
  return newSymbol(sym.QNAME, yytext());
}
return newSymbol(ss);
}

/**
 * If symbol is first token or if it follows any of the operators
 * listed in http://www.w3.org/TR/xpath#exprlex then treat as a
 * wildcard instead of a multiplication operator
 */
Symbol disambiguateStar() throws Exception {
  switch (last) {
    case -1:  // first token
    case sym.ATSIGN:
    case sym.DCOLON:
    case sym.LPAREN:
    case sym.LBRACK:
    case sym.COMMA:
    case sym.AND:
    case sym.OR:
    case sym.MOD:
    case sym.DIV:
    case sym.MULT:
    case sym.SLASH:
    case sym.DSLSH:
    case sym.VBAR:
    case sym.PLUS:
    case sym.MINUS:
    case sym.EQ:
    case sym.NE:
    case sym.LT:
    case sym.LE:
case sym.GT:
    case sym.GE:
        return newSymbol(sym.STAR);
    }
return newSymbol(sym.MULT);
}

Symbol newSymbol(int ss) {
    last = ss;
    return new Symbol(ss);
}

Symbol newSymbol(int ss, String value) {
    last = ss;
    return new Symbol(ss, value);
}

Symbol newSymbol(int ss, Long value) {
    last = ss;
    return new Symbol(ss, value);
}

Symbol newSymbol(int ss, Double value) {
    last = ss;
    return new Symbol(ss, value);
}
%
%eofval{
return newSymbol(sym.EOF);
%eofval}
%yylexthrow{
    Exception
%yylexthrow}

Letter={[BaseChar]|[Ideographic]}

BaseChar=[u0041-u005A\u0061-u007A\u00C0-u00D6\u00D8-u00F6-u00F8-u00FF\u0100-u0131\u0134-
\u013E\u0141-u0148\u014A\u017E\u0180-u01C3\u01CD\u01F0\u01F4\u01F5\u01FA-u0217\u0250-
\u02A8\u02BB-\u02C1\u0366\u0368-\u0384\u0386\u0388-\u038AU038CU038E\u03A1\u03A3-\u03CEu03D0-
\u03D6\u03DA\u03DE\u03E0\u03F3\u0401-\u040C\u040E-\u044Fu0451-\u045Cu045E-
\u0481\u0490-\u04C7-\u04C8\u04CB\u04CC\u04D0-\u04E8\u04F8\u0531-
\u0556\u0559\u0561-\u0586\u05D0-\u05EA\u05F0-\u05FA\u0621-\u0641-\u0661-\u0663\u0671-\u06B7\u06BA-
\u06EB\u06CA\u06CE\u06D0-\u06D3\u06D5\u06E5-\u06E6\u0905-\u0939\u093D\u0958-\u0961-\u0985-
\u0988\u0993-\u09A8\u09AA-\u09B0-\u09B2-\u09B6-\u09B9-\u09DC-\u09DD-\u09DF-
\u09E1-\u09F0-\u09F1-\u0A05-\u0A0A-\u0A0F-\u0A10-\u0A13-\u0A28-\u0A2A-\u0A30-\u0A32-\u0A33-u0A35-
\u0A36-\u0A38-\u0A39-\u0A59-\u0A5C-\u0A5E-\u0A72-\u0A74-\u0A85-\u0A8B-\u0A8D-\u0A8F-\u0A91-\u0A93-
HighSurrogate=[\uD800-\uDBFF]
LowSurrogate=[\uDC00-\uDFFF]

```java
/**
 * return disambiguateStar();
 */

/**
 * return newSymbol(sym.SLASH);
 */

/**
 * return newSymbol(sym.PLUS);
 */

/**
 * return newSymbol(sym.MINUS);
 */

/**
 * return disambiguateOperator(sym.DIV);
 */

/**
 * return disambiguateOperator(sym.MOD);
 */

/**
 * return newSymbol(sym.DCOLON);
 */

/**
 * return newSymbol(sym.COMMA);
 */

/**
 * return newSymbol(sym.ATSIGN);
 */

/**
 * return newSymbol(sym.DDOT);
 */

/**
 * return newSymbol(sym.VBAR);
 */

/**
 * return newSymbol(sym.DOLLAR);
 */

/**
 * return newSymbol(sym.DSLASH);
 */

/**
 * return newSymbol(sym.EQ);
 */

/**
 * return newSymbol(sym.NE);
 */

/**
 * return newSymbol(sym.LT);
 */

/**
 * return newSymbol(sym.GT);
 */

/**
 * return newSymbol(sym.LE);
 */

/**
 * return newSymbol(sym.GE);
 */

/**
 * return disambiguateAxisOrFunction(sym.ID);
 */

/**
 * return disambiguateAxisOrFunction(sym.KEY);
 */

/**
 * return newSymbol(sym.TEXT);
 */

/**
 * return newSymbol(sym.TEXT);
 */

/**
 * return newSymbol(sym.NODE);
 */

/**
 * return newSymbol(sym.NODE);
 */

/**
 * return newSymbol(sym.COMMENT);
 */

/**
 * return newSymbol(sym.COMMENT);
 */

/**
 * return disambiguateAxisOrFunction(sym.PIPARAM);
 */

/**
 * return newSymbol(sym.PI);
 */

/**
 * return newSymbol(sym.PI);
 */

/**
 * return disambiguateOperator(sym.OR);
 */

/**
 * return disambiguateOperator(sym.AND);
 */

/**
 * return disambiguateAxisOrFunction(sym.CHILD);
 */

/**
 * return disambiguateAxisOrFunction(sym.ATTRIBUTE);
 */

/**
 * return disambiguateAxisOrFunction(sym.ANCESTOR);
 */

/**
 * return disambiguateAxisOrFunction(sym.ANCESTORORSELF);
 */

/**
 * return disambiguateAxisOrFunction(sym.DESCENDANT);
 */

/**
 * return disambiguateAxisOrFunction(sym.DESCENDANTORSELF);
 */

/**
 * return disambiguateAxisOrFunction(sym.FOLLOWING);
 */

/**
 * return disambiguateAxisOrFunction(sym.FOLLOWINGSIBLING);
 */

/**
 * return disambiguateAxisOrFunction(sym.NAMESPACE);
 */

/**
 * return disambiguateAxisOrFunction(sym.PARENT);
 */

Open Source Used in DNAC Platform Arica 3824
"preceding" { return disambiguateAxisOrFunction(sym.PRECEDING); }
"preceding-sibling" { return disambiguateAxisOrFunction(sym.PRECEDINGSIBLING); }
"self" { return disambiguateAxisOrFunction(sym.SELF); }
"[" { return newSymbol(sym.LBRACK); }
"]" { return newSymbol(sym.RBRACK); }
"(" { return newSymbol(sym.LPAREN); }
")" { return newSymbol(sym.RPAREN); }
"<PATTERN>" { initialize(); return new Symbol(sym.PATTERN); }
"<EXPRESSION>" { initialize(); return new Symbol(sym.EXPRESSION); }
\"[^\\"]*\" { return newSymbol(sym.Literal, yytext().substring(1, yytext().length() - 1)); }
\[\]^\[\] { return newSymbol(sym.Literal, yytext().substring(1, yytext().length() - 1)); }
{Digit}+ { return newSymbol(sym.INT, new Long(yytext())); }
{Digit}+("[Digit]*")? { return newSymbol(sym.REAL, new Double(yytext())); }
"."{Digit}+ { return newSymbol(sym.REAL, new Double(yytext())); }
"." { return newSymbol(sym.DOT); }
{NCName}":"?[NCName] { return newSymbol(sym.QNAME, yytext()); }
{NCName}":"*" { return newSymbol(sym.QNAME, yytext()); }
{NCName}":"*@" { return newSymbol(sym.QNAME, yytext()); }
[ \t\n\f] { /* ignore white space. */ }
. { throw new Exception(yytext()); }

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources.jar/org/apache/xalan/xsltc/compiler/xpath.lex
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
*/
* $Id: XSLProcessorVersion.src 468640 2006-10-28 06:53:53Z minchau $
package org.apache.xalan.processor;

/**
 * Administrative class to keep track of the version number of
 * the Xalan release.
 * <P>See also: org/apache/xalan/res/XSLTInfo.properties</P>
 * @deprecated To be replaced by org.apache.xalan.Version.getVersion()
 * @xsl.usage general
 */
public class XSLProcessorVersion
{

/**
 * Print the processor version to the command line.
 * @param argv command line arguments, unused.
 */
public static void main(String argv[])
{
    System.out.println(S_VERSION);
}

/**
 * Constant name of product.
 */
public static final String PRODUCT = "Xalan";

/**
 * Implementation Language.
 */
public static final String LANGUAGE = "Java";

/**
 * Major version number.
 * Version number. This changes only when there is a
 * significant, externally apparent enhancement from
 * the previous release. 'n' represents the n'th
 * version.
 *
 * Clients should carefully consider the implications
 * of new versions as external interfaces and behaviour
 * may have changed.
 */
public static final int VERSION = @version.VERSION@;

/**
 * Release Number.
 * Release number. This changes when:
* - a new set of functionality is to be added, eg,
  * implementation of a new W3C specification.
  * - API or behaviour change.
  * - its designated as a reference release.

```java
public static final int RELEASE = @version.RELEASE@;
```

/**
 * Optional identifier used to designate maintenance
 * drop applied to a specific release and contains
 * fixes for defects reported. It maintains compatibility
 * with the release and contains no API changes.
 * When missing, it designates the final and complete
 * development drop for a release.
 */

```java
public static final int MAINTENANCE = @version.MINOR@;
```

/**
 * Optional identifier designates development drop of
 * a specific release. D01 is the first development drop
 * of a new release.
 *
 * Development drops are works in progress towards a
 * completed, final release. A specific development drop
 * may not completely implement all aspects of a new
 * feature, which may take several development drops to
 * complete. At the point of the final drop for the
 * release, the D suffix will be omitted.
 *
 * Each 'D' drops can contain functional enhancements as
 * well as defect fixes. 'D' drops may not be as stable as
 * the final releases.
 */

```java
public static final int DEVELOPMENT = 0;
```

/**
 * Version String like <CODE>"<B>Xalan</B> <B>Language</B> <b>v.r[.dd| <B>D</B>nn"]</CODE>.
 * <p> Semantics of the version string are identical to the Xerces project.</p>
 */

```java
public static final String S_VERSION = PRODUCT+" "+LANGUAGE+" "
+VERSION+"."+RELEASE+"."
+(DEVELOPMENT > 0 ? ("D"+DEVELOPMENT) 
: (""+MAINTENANCE));
```
No license file was found, but licenses were detected in source scan.

* Licensed to the Apache Software Foundation (ASF) under one
  * or more contributor license agreements. See the NOTICE file
  * distributed with this work for additional information
  * regarding copyright ownership. The ASF licenses this file
  * to you under the Apache License, Version 2.0 (the "License");
  * you may not use this file except in compliance with the License.
  * You may obtain a copy of the License at
  *
  *     http://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  * See the License for the specific language governing permissions and
  * limitations under the License.

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/trace/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/utils/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/operations/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/patterns/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/serialized/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsl/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/xsl/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/xsl/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/axes/package.html
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/res/package.html
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

// Proprietary, built in functions
/** current function string (Proprietary). */

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/compiler/Keywords.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/functions/Function.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/trax/XSLTSource.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/axes/AttributeIterator.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/dom/LoadDocument.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/runtime/ObjectFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorExsltFuncResult.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/If.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/util/ErrorMessages_zh.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/If.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/util/PrefixResolver.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/XSLTErrorResources_en.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/PrefixResolver.java

---

**Open Source Used In DNAC Platform Artica**

---

3831
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/res/StringArrayWrapper.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Choose.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/ElemFallback.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/transformer/XalanProperties.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Stylesheet.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ErrorMessages_zh_TW.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/IntExpr.java

---

Open Source Used In DNAC Platform Arica 3832
jAR/Org/apache/xml/dtm/ref/IncrementalSAXSource_Xerces.java
* JAR/Org/apache/xml/utils/res/XResources_ko.java
* JAR/Org/apache/xalan/xslt/dom/NodeSortRecord.java
* JAR/Org/apache/xalan/xslt/compiler/util/RtMethodGenerator.java
* JAR/Org/apache/xalan/xslt/compiler/util/ErrorMessages_ca.java
* JAR/Org/apache/xalan/xslt/runtime/BasisLibrary.java
* JAR/Org/apache/xalan/xslt/runtime/Parameter.java
* JAR/Org/apache/xalan/xslt/compiler/util/ErrorMessages_pl.java
* JAR/Org/apache/xalan/xslt/runtime/NodeSortRecord.java
* JAR/Org/apache/xml/dtm/ref/DTMSafeStringPool.java
* JAR/Org/apache/xml/serializer/DOM3Serializer.java
* JAR/Org/apache/xml/utils/SuballocatedIntVector.java

Open Source Used In DNAC Platform Arica 3836
Open Source Used In DNAC Platform Arica
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/operations/NotEquals.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/ExsltSets.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/runtime/ErrorMessages_ko.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/XMLErrorResources.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/SecuritySupport.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/ExtensionNamespacesManager.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/res/XPATHErrorResources_fr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/ElemChoose.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/sql/DefaultConnectionPool.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/MethodResolver.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Constants.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/utils/MsgKey.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Import.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/dom/StripWhitespaceFilter.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/SymbolTable.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/utils/URI.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/utils/SystemIDResolver.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/util/ErrorMsg.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/res/XPATHErrorResources_ja.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/UnionPathExpr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/cmdline/getopt/IllegalArgumentException.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/SerializerTrace.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/XMLErrorResources_hu.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/transformer/DecimalToRoman.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/Hashtree2Node.java
Open Source Used In DNAC Platform Arica 3844

* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/dom/SAXImpl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/dom/DocumentCache.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/ExtensionHandler.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorTemplate.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/operations/Lte.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/sql/SQLErrorDocument.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/ElemMessage.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/LocaleUtility.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/XSLProcessorContext.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/ref/ChunkedIntArray.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/DTMAxisIterator.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/FunctionAvailableCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/ref/DTMDocumentImpl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/StringLengthCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/operations/VariableSafeAbsRef.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorText.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/XSLTErrorResources_fr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/extensions/ExpressionVisitor.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/AVTPartSimple.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/ToUnknownStream.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/util/NodeSortRecordFactGenerator.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/util/OutlineableChunkEnd.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/CharInfo.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/XMLErrorResources_sl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/LiteralAttribute.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/jaxp/XPathImpl.java
Open Source Used In DNAC Platform Arica 3851

* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/utils/SerializerMessages_cs.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/FuncKey.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ApplyTemplates.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ParentPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Appliques.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/RelativePathPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/UnresolvedRef.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ParentPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ObjectFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/dom/SortSettings.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ToString.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Traversal.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ApplyTemplates.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/FunctionCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/UnresolvedRef.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ParentPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ObjectFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ToString.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Traversal.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ApplyTemplates.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/FunctionCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/UnresolvedRef.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ParentPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ObjectFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ToString.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/Traversal.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ApplyTemplates.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/FunctionCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/UnresolvedRef.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/ParentPattern.java
Open Source Used In DNAC Platform Arica

jar/org/apache/xalan/xslt/compiler/LastCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/utils/BoolStack.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/VariableBase.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/NodeCounter.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/ProcessingInstructionPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/Translet.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/ObjectStack.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/templates/ElemTemplate.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/dom/NodeCounter.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/trax/SAX2DOM.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/ExpressionOwner.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/IdPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/EXMLErrorResources_sv.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/runtime/ErrorMsgMessages_fr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/objects/XBooleanStatic.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/trax/TransformerImpl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/SyntaxTreeNode.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorImport.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/axes/HasPositionalPredChecker.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/functions/FuncCount.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/ref/ExtendedType.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Utilities.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Direction.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/ProcessingInstructionPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/Translet.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/ObjectStack.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/IdPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/EXMLErrorResources_sv.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/runtime/ErrorMsgMessages_fr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/objects/XBooleanStatic.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/trax/TransformerImpl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/SyntaxTreeNode.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorImport.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/axes/HasPositionalPredChecker.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/functions/FuncCount.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/dtm/ref/ExtendedType.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Utilities.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Direction.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/Translet.java
jar/org/apache/xalan/xslt/runtime/Constants.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xml/dtm/ref/sax2dtm/SAX2DTM.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xml/utils/SAXSourceLocator.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/compiler/util/MarkerInstruction.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/CollatorFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/res/XSLTErrorResources_sk.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/compiler/util/ErrorMessages.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/compiler/util/UnsupportedElement.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/cmdline/ObjectFactory.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/dom/EmptyFilter.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/extensions/ExtensionHandlerExsltFunction.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/Runtime/Node.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/lib/sql/DTMDocument.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-
jar/org/apache/xalan/xsltc/cmdline/Transform.java

Open Source Used In DNAC Platform Arica 3859
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/res/XResources_en.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/dom/SortingIterator.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/compiler/XPathParser.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/serializer/WriterToUTF8Buffered.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/NotCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xsltc/compiler/StringCall.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/ExsltCommon.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/lib/sql/XConnection.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/functions/SecuritySupport.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/res/XSLTErrorResources JA.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorExsltFunction.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/res/XPATHErrorResources_ko.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/processor/ProcessorUnknown.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/domapi/XPathNamespaceImpl.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/operations/Mod.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/AbsolutePathPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Variable.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/utils/AttList.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/FilterExpr.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/axes/ReverseAxesWalker.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/patterns/UnionPattern.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xml/res/XMLErrorResources JA.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xpath/XPathContext.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/compiler/Include.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/templates/ElemWithParam.java
* /opt/cola/permits/1134796662_1613422294.09/0/xalan-2-7-2-sources-jar/org/apache/xalan/xslt/dom/CurrentNodeListFilter.java

Open Source Used In DNAC Platform Arica 3860
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation.
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1073 side-channel 1.0.4

1.1073.1 Available under license :
MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1074 got 9.6.0
1.1074.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1075 string-width 4.2.2
1.1075.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1076 strip-ansi 6.0.0 6.0.0
1.1076.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1077 jcl-over-slf4j 1.7.26

1.1078 normalize-path 2.1.1
1.1078.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1079 babel-preset-stage-3 6.24.1

1.1079.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
liable FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1080 @babel/helper-validator-option 7.14.5
1.1080.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1081 bootstrap 4.3.1 4.3.1
1.1081.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011-2019 Twitter, Inc.
Copyright (c) 2011-2019 The Bootstrap Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1082 jsonwebtoken 7.4.3 7.4.3
1.1082.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Auth0, Inc. <support@auth0.com> (http://auth0.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1083 moment 2.24.0 2.24.0
1.1083.1 Available under license :

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.1084 jersey-entity-filtering 2.25.1
1.1084.1 Available under license :
   CDDL+GPL License
   http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1085 has-symbols 1.0.0
1.1085.1 Available under license :
   MIT License
   Copyright (c) 2016 Jordan Harband
   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all
   copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
   SOFTWARE.

1.1086 guava 30.0-jre
1.1086.1 Available under license :
   No license file was found, but licenses were detected in source scan.
/*
* Copyright (C) 2011 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Verify.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractTable.java

/*
* Copyright (C) 2013 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Verify.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractTable.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2005 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ClosingFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/Traverser.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Platform.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Internal.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Platform.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Internal.java
No license file was found, but licenses were detected in source scan.
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AbstractService.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/MapMaker.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/MapMaker.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/DenseImmutableTable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Splitter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/Cut.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/RegularImmutableTable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/MapMakerInternalMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/xml/XmlEscapers.java

No license file was found, but licenses were detected in source scan.

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AtomicDoubleArray.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of {code Doubles}. Intended to be empty for regular
 * version.
 */

Found in path(s):

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ForwardingSetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ForwardingImmutableCollection.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/MinMaxPriorityQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/AbstractSequentialIterator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ForwardingListMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/RowSortedTable.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/CompactHashing.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/IncidentEdgeSet.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 */
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This method was rewritten in Java from an intermediate step of the Murmur hash function in
http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
following header:

MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
hereby disclaims copyright to this source code.

odied in path(s):

/opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Hashing.java
No license file was found, but licenses were detected in source scan.

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License
is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing permissions and limitations under
the License.

This following method is a modified version of one found in
http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
which contained the following notice:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
*/

Found in path(s):
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2011 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
* either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractRangeSet.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Count.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/RangeSet.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ForwardingSortedMultiset.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/SortedIterables.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/GeneralRange.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/RegularImmutableSortedMultiset.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/SortedIterable.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ImmutableSortedMultiset.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
 *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/package-info.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/CollectSpliterators.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ImmutableBiMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2017 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
 *
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/ImmutableLongArray.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/ImmutableDoubleArray.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/ImmutableIntArray.java
 * /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/AbstractHashFunction.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Outer class that exists solely to let us write [@code Partially.GwtIncompatible] instead of plain
 * [@code GwtIncompatible]. This is more accurate for [@link Futures#catching], which is available
 * under GWT but with a slightly different signature.
 *
 * <p>We can't use [@code PartiallyGwtIncompatible] because then the GWT compiler wouldn't recognize
 * it as a [@code GwtIncompatible] annotation. And for [@code Futures.catching], we need the GWT
 * compiler to autostrip the normal server method in order to expose the special, inherited GWT
 * version.
 */

Found in path(s):
/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 */
* the License.
*/

Found in path(s):  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/UncheckedExecutionException.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AbstractScheduledService.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/Hashing.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Queues.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/TreeRangeSet.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/UnsignedLong.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/Murmur3_32HashFunction.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/AbstractStreamingHasher.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/UnsignedInteger.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/AbstractCache.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/DescendingImmutableSortedMultiset.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/RegularContiguousSet.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/MathPreconditions.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/package-info.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ListeningScheduledExecutorService.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/RemovalNotification.java  
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/HttpHeaders.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Present.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/Funnel.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/MessageDigestHashFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/ForwardingLoadingCache.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ImmutableTable.java
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/SingletonImmutableTable.java
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ImmutableEnumSet.java
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/AbstractIndexedListIterator.java
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ImmutableClassToInstanceMap.java
  /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/ImmutableMultiset.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * Returns an array containing all of the elements in the specified collection. This method
 * returns the elements in the order they are returned by the collection's iterator. The returned
 * array is “safe” in that no references to it are maintained by the collection. The caller is
 * thus free to modify the returned array.
 *
 * <p>This method assumes that the collection size doesn't change while the method is running.
 *
 * <p>TOD(kevinb): support concurrently modified collections?
 */
* @param c the collection for which to return an array of elements
*/

Found in path(s):
*/opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ObjectArrays.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2011 The Guava Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
*/opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2018 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
*/opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/ImmutableSupplier.java
*/opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ExecutionSequencer.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/JdkBackedImmutableSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/JdkBackedImmutableMultiset.java

No license file was found, but licenses were detected in source scan.

/**
   * Copyright (C) 2020 The Guava Authors
   *
   * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
   * in compliance with the License. You may obtain a copy of the License at
   * *
   * http://www.apache.org/licenses/LICENSE-2.0
   *
   * Unless required by applicable law or agreed to in writing, software distributed under the License
   * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
   * or implied. See the License for the specific language governing permissions and limitations under
   * the License.
   */

* Holder for web specializations of methods of [@code Ints]. Intended to be empty for regular
  * version.
* /

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/IntsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/**
   * Copyright (C) 2012 The Guava Authors
   *
   * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
   *
   * http://www.apache.org/licenses/LICENSE-2.0
   *
   * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
   */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/RangeMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ForwardingImmutableMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/TransformedIterator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ImmutableEnumMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/SortedMultisetBridge.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/EvictingQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/AllEqualOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ForwardingNavigableMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/CompactHashSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ForwardingImmutableList.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/DescendingMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/FilteredEntrySetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ForwardingImmutableSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/TreeRangeMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ForwardingNavigableSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/CompactLinkedHashMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/FilteredSetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/FilteredEntryMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/TransformedListIterator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/CompactHashMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/ForwardingDeque.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/DescendingImmutableSortedSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/TreeTraverser.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/AbstractNavigableMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/CompactLinkedHashSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/AbstractMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/FilteredMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/ForwardingBlockingDeque.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/
/
* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
* following header:
*
* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
* hereby disclaims copyright to this source code.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ImmutableSortedAsList.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/LinkedHashMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Multimaps.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Multisets.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ReverseNaturalOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/TreeMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/HashMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Iterators.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Maps.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/LinkedListMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ForwardingConcurrentMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/EnumBiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/NaturalOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractMapEntry.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ImmutableList.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ForwardingObject.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Iterables.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Lists.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/SortedSetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ListMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ConcurrentHashMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Sets.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ForwardingMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/NullsLastOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/AbstractBiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/Multiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ReverseOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/BiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ForwardingMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ForwardingSortedSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ByFunctionOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/SetMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/TreeMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/Multimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/EnumHashBiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/AbstractIterator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/Ordering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ForwardingSortedMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/MapDifference.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/CompoundOrdering.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/AbstractMapBasedMultiset.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collection/ForwardingMapEntry.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
/*

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/annotations/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Ascii.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Equivalence.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/annotations/Beta.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Strings.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ListeningExecutorService.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/Stats.java
jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/reflect/Element.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/io/Closer.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/html/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/xml/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/io/BaseEncoding.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/TopKSelector.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/RecursiveDeleteOption.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/Network.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/Graphs.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/graph/Graph.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2008 The Guava Authors
*Licensed under the Apache License, Version 2.0 (the "License");
*you may not use this file except in compliance with the License.
*You may obtain a copy of the License at
*
*http://www.apache.org/licenses/LICENSE-2.0
*
*Unless required by applicable law or agreed to in writing, software
*distributed under the License is distributed on an "AS IS" BASIS,
*WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*See the License for the specific language governing permissions and
*limitations under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you
 * may not use this file except in compliance with the License. You may
 * obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
 * implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/TreeBasedTable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/CollectPreconditions.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/Range.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/ImmutableEntry.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collection/StandardRowSortedTable.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you
 * may not use this file except in compliance with the License. You may
 * obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
 * implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */
Open Source Used In DNAC Platform Arica 3912

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/JdkBackedImmutableBiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/IndexedImmutableSet.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/BaseImmutableMultimap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2006 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/FluentFuture.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/Futures.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/escape/CharEscaperBuilder.java

Open Source Used In DNAC Platform Arica 3913
Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collection/SortedMultiset.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/*
 * This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
 * and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 * As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
 * java.util.HashMap class.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 */
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Doubles.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Chars.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Longs.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Booleans.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/net/InetAddresses.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/Converter.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/base/CharMatcher.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/escape/Escaper.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/Ints.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/FluentIterable.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/primitives/UnicodeEscaper.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/thirdparty/publicsuffix/TrieParser.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/SequentialExecutor.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/MultiReader.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * Not supported. <b>You are attempting to create a map that may contain a non-{@code Comparable} key.</b> Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy version.
 *
 * @throws UnsupportedOperationException always
 * @deprecated <b>Pass a key of type {@code Comparable} to use {@link ImmutableSortedMap#of(Comparable, Object)}.\</b>
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
* See the License for the specific language governing permissions and 
* limitations under the License.
*/

Found in path(s): 
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/AbstractGraphBuilder.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/EndpointPair.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/StandardValueGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/MoreCollectors.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/CollectCollectors.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/collect/LinkedHashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/UndirectedNetworkConnections.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-
jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2020 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/Java8Compatibility.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/BigDecimalMath.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/Java8Compatibility.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/math/ToDoubleRounder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/io/Resources.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/primitives/Primitives.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/ByteStreams.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/Interners.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources.jar/com/google/common/base/Throwables.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
*/
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

/*

* Copyright (C) 2015 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
* the License.
*/

Found in path(s):
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/Platform.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/AggregateFutureState.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1102081303_1603450661.12/0/guava-30-0-jre-sources-jar/com/google/common/hash/LittleEndianByteArray.java

1.1087 @babel/helper-wrap-function 7.16.0
1.1087.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
1.1088 jsesc 0.5.0
1.1088.1 Available under license :
Copyright Mathias Bynens <http://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1089 activation-api 1.2.1
1.1089.1 Available under license :
# Notices for Eclipse Project for JAF

This content is produced and maintained by the Eclipse Project for JAF project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaf

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms...
of the Eclipse Distribution License v. 1.0, which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaf

## Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1090 protobuf-java 3.11.0

1.1090.1 Available under license:

    // Redistribution and use in source and binary forms, with or without
    // modification, are permitted provided that the following conditions are
    // met:
    //
    // * Redistributions of source code must retain the above copyright
    // notice, this list of conditions and the following disclaimer.
    // * Redistributions in binary form must reproduce the above
    // copyright notice, this list of conditions and the following disclaimer
    // in the documentation and/or other materials provided with the
    // distribution.
    // * Neither the name of Google Inc. nor the names of its
    // contributors may be used to endorse or promote products derived from
    // this software without specific prior written permission.
    //
    // THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
    // "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
    // LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
    // A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
    // OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
    // SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
    // LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
    // DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
    // THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
    // (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
    // OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1091 @babel/preset-env 7.16.4

1.1091.1 Available under license:

    MIT License

    Copyright (c) 2014-present Sebastian McKenzie and other contributors

    Permission is hereby granted, free of charge, to any person obtaining
    a copy of this software and associated documentation files (the
    "Software"), to deal in the Software without restriction, including
    without limitation the rights to use, copy, modify, merge, publish,
    distribute, sublicense, and/or sell copies of the Software, and to
    permit persons to whom the Software is furnished to do so, subject to
    the following conditions:

    The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1092 lodash-omit 4.5.0 4.5.0

1.1092.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as
documented below:

====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1093 commons-codec 1.13
1.1093.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Codec

Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.
1.1094 nimbus-jose-jwt 7.9
1.1094.1 Available under license:
Nimbus JOSE + JWT

Copyright 2012 - 2018, Connect2id Ltd.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use
this file except in compliance with the License. You may obtain a copy of the
License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed
under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the
specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1095 cups 2.1.3 4ubuntu0.10
1.1095.1 Available under license:

<html>
<!-- SECTION: Getting Started -->
<head>
<title>Apache License Version 2.0</title>
<link rel="stylesheet" type="text/css" href="../cups-printable.css">
</head>
<body>

<h1>Apache License</h1>

<p>Version 2.0, January 2004<br />
http://www.apache.org/licenses/LICENSE-2.0</p>

<p>TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION</p>

<p><strong>1. Definitions</strong></p>
License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You
a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly
display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.</p>
<p><strong>a name="patent">3. Grant of Patent License</a></strong>. Subject to the terms
and conditions of this License, each Contributor hereby grants to You a
perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use,
offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor
that are necessarily infringed by their Contribution(s) alone or by
combination of their Contribution(s) with the Work to which such
Contribution(s) was submitted. If You institute patent litigation against
any entity (including a cross-claim or counterclaim in a lawsuit) alleging
that the Work or a Contribution incorporated within the Work constitutes
direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the
date such litigation is filed.</p>
<p><strong>a name="redistribution">4. Redistribution</a></strong>. You may reproduce and
distribute copies of the Work or Derivative Works thereof in any medium,
with or without modifications, and in Source or Object form, provided that
You meet the following conditions:</p>
<ol style="list-style: lower-latin;">
<li>You must give any other recipients of the Work or Derivative Works a
copy of this License; and</li>
<li>You must cause any modified files to carry prominent notices stating
that You changed the files; and</li>
<li>You must retain, in the Source form of any Derivative Works that You
distribute, all copyright, patent, trademark, and attribution notices from
the Source form of the Work, excluding those notices that do not pertain to
any part of the Derivative Works; and</li>
<li>If the Work includes a "NOTICE" text file as part of its distribution,
then any Derivative Works that You distribute must include a readable copy
of the attribution notices contained within such NOTICE file, excluding
those notices that do not pertain to any part of the Derivative Works, in
at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation,
if provided along with the Derivative Works; or, within a display generated
by the Derivative Works, if and wherever such third-party notices normally
appear. The contents of the NOTICE file are for informational purposes only
and do not modify the License. You may add Your own attribution notices
within Derivative Works that You distribute, alongside or as an addendum to
the NOTICE text from the Work, provided that such additional attribution
notices cannot be construed as modifying the License.</li>
</ol>
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf.
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.\</p>
<p>END OF TERMS AND CONDITIONS</p>
<h2>APPENDIX: How to apply the Apache License to your work</h2>
<p>To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.\</p>
<pre>
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
</pre>
<h2>CUPS Exceptions to the Apache 2.0 License</h2>
<p>As an exception, if, as a result of your compiling your source code, portions of this Software are embedded into an Object form of such source code, you may redistribute such embedded portions in such Object form without complying with the conditions of Sections 4(a), 4(b) and 4(d) of the License.\</p>
<p>In addition, if you combine or link compiled forms of this Software with software that is licensed under the GPLv2 ("Combined Software") and if a court of competent jurisdiction determines that the patent provision (Section 3), the indemnity provision (Section 9) or other Section of the License conflicts with the conditions of the GPLv2, you may retroactively and prospectively choose to deem waived or otherwise exclude such Section(s) of the License, but only in their entirety and only with respect to the Combined Software.\</p>
</BODY>
</HTML>
Copyright 1992, 1993, 1994, 1997 Henry Spencer. All rights reserved. This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on any computer system, and to alter it and redistribute it, subject to the following restrictions:

1. The author is not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources, credits must appear in the documentation.

3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.

4. This notice may not be removed or altered.

CUPS

Copyright 2007-2019 by Apple Inc.

Copyright 1997-2007 by Easy Software Products.

CUPS and the CUPS logo are trademarks of Apple Inc.

The MD5 Digest code is Copyright 1999 Aladdin Enterprises.

The Kerberos support code ("KSC") is copyright 2006 by Jelmer Vernooij and is provided 'as-is', without any express or implied warranty. In no event will the author or Apple Inc. be held liable for any damages arising from the use of the KSC.

Sources files containing KSC have the following text at the top of each source file:

This file contains Kerberos support code, copyright 2006 by Jelmer Vernooij.

The KSC copyright and license apply only to Kerberos-related feature code in CUPS. Such code is typically conditionally compiled based on the present of the HAVE_GSSAPI preprocessor definition.

Permission is granted to anyone to use the KSC for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of the KSC must not be misrepresented; you must not claim that
you wrote the original software. If you use the KSC in a product, an
acknowledgment in the product documentation would be appreciated but is not
required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

-- CUPS Exceptions to the Apache 2.0 License --

As an exception, if, as a result of your compiling your source code, portions
of this Software are embedded into an Object form of such source code, you
may redistribute such embedded portions in such Object form without complying
with the conditions of Sections 4(a), 4(b) and 4(d) of the License.

In addition, if you combine or link compiled forms of this Software with
software that is licensed under the GPLv2 ("Combined Software") and if a
court of competent jurisdiction determines that the patent provision (Section
3), the indemnity provision (Section 9) or other Section of the License
conflicts with the conditions of the GPLv2, you may retroactively and
prospectively choose to deem waived or otherwise exclude such Section(s) of
the License, but only in their entirety and only with respect to the Combined
Software.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1096 jersey-media-multipart 2.25.1
1.1096.1 Available under license :
     CDDL+GPL License
     http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1097 string-width 1.0.2 1.0.2
1.1097.1 Available under license :
     The MIT License (MIT)

     Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1098 jsreport-freeze 2.0.0 2.0.0

1.1098.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1099 babel-plugin-polyfill-corejs2 0.3.0

1.1099.1 Available under license:

MIT License

Copyright (c) 2014-present Nicol Ribaudo and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1100 protobuf 2.6.1-1.3

1.1100.1 Available under license:

This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:
Sanjay Ghemawat <sj@google.com>
Jeff Dean <jeff@google.com>
Daniel Dulitz <daniel@google.com>
Craig Silverstein
Paul Haahr <haahr@google.com>
Corey Anderson <corin@google.com>
(and many others)

Proto2 C++ and Java primary author:
Kenton Varda <kenton@google.com>

Proto2 Python primary authors:
Will Robinson <robinson@google.com>
Petar Petrov <petar@google.com>

Large code contributions:
Jason Hsueh <jasonh@google.com>
Joseph Schorr <jschorr@google.com>
Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:
Scott Bruce <sbruce@google.com>
Frank Yellin
Neal Norwitz <nnorwitz@google.com>
Jeffrey Yasskin <jyasskin@google.com>
Ambrose Feinstein <ambrose@google.com>

Documentation:
Lisa Carey <lcarey@google.com>

Maven packaging:
Gregory Kick <gak@google.com>

Patch contributors:
Kevin Ko <kevin.s.ko@gmail.com>
   * Small patch to handle trailing slashes in --proto_path flag.
Johan Euphrosine <proppy@aminche.com>
   * Small patch to fix Python CallMethod().
Ulrich Kunitz <kune@deine-taler.de>
   * Small optimizations to Python serialization.
Leandro Lucarella <llucax@gmail.com>
   * VI syntax highlighting tweaks.
   * Fix compiler to not make output executable.
Dilip Joseph <dilip.antony.joseph@gmail.com>
   * Heuristic detection of sub-messages when printing unknown fields in text format.
Brian Atkinson <nairb774@gmail.com>
   * Added @Override annotation to generated Java code where appropriate.
Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>
   * Tru64 support.
Monty Taylor <monty.taylor@gmail.com>
   * Solaris 10 + Sun Studio fixes.
Alek Storm <alek.storm@gmail.com>
   * Slicing support for repeated scalar fields for the Python API.
Oleg Smolsky <oleg.smolsky@gmail.com>
   * MS Visual Studio error format option.
   * Detect unordered_map in stl_hash.m4.
Brian Olson <brianolson@google.com>
   * gzip/zlib I/O support.
Michael Poole <mdpoole@troilus.org>
   * Fixed warnings about generated constructors not explicitly initializing
all fields (only present with certain compiler settings).
* Added generation of field number constants.
Wink Saville <wink@google.com>
* Fixed initialization ordering problem in logging code.
Will Pierce <willp@nuclei.com>
* Small patch improving performance of in Python serialization.
Alexandre Vassalotti <alexandre@peadrop.com>
* Emacs mode for Protocol Buffers (editors/protobuf-mode.el).
Scott Stafford <scott.stafford@gmail.com>
* Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.
Alexander Melnikov <alm@sibmail.ru>
* HPUX support.
Oliver Jowett <oliver.jowett@gmail.com>
  * Detect whether zlib is new enough in configure script.
  * Fixes for Solaris 10 32/64-bit confusion.
Evan Jones <evanj@mit.edu>
  * Optimize Java serialization code when writing a small message to a stream.
  * Optimize Java serialization of strings so that UTF-8 encoding happens only
    once per string per serialization call.
  * Clean up some Java warnings.
  * Fix bug with permanent callbacks that delete themselves when run.
Michael Kucharski <m.kucharski@gmail.com>
  * Added CodedInputStream.getTotalBytesRead().
Kacper Kowalik <xarthisius.kk@gmail.com>
  * Fixed m4/acx_pthread.m4 problem for some Linux distributions.
William Orr <will@worbase.com>
  * Fixed detection of sched_yield on Solaris.
  * Added atomicops for Solaris
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Testing Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@google.com>
Ji Sigursson <joi@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@google.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@google.com>
Patrick Riley <pfr@google.com>
Peter Kaminski <piotrkgoogle.com>
Preston Jackson <preston.a.jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@google.com>
Russ Rufer <russ@pentad.com>
Sean McAfee <eefacm@gmail.com>
Sigurur Sgeirsson <siggigoogle.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@google.com>
Zhanyong Wan <wan@google.com>

1.1101 jctools-core 2.1.1
1.1101.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/BaseLinkedQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/SpmcAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/MpscChunkedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/maps/NonBlockingHashMap.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/util/UnsafeAccess.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/LinkedQueueNode.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/MpscChunkedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpscLinkedAtomicQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MessagePassingQueueUtil.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpchunckedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/SpscAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/MpscGrowableArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/ConcurrentCircularArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpmcArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/util/RangeUtil.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/LinkedQueueAtomicNode.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/SequencedAtomicReferenceArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/spec/Preference.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/spec/Ordering.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscLinkedQueue7.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/BaseMpscLinkedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscLinkedQueue8.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/maps/NonBlockingIdentityHashMap.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/maps/QueueFactory.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/maps/NonBlockingHashSet.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscGrowableArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MessagePassingQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/util/Pow2.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/MpscUnboundedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/BaseSpscLinkedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/BaseMpscLinkedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpscGrowableArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpmcArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/spec/ConcurrentQueueSpec.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/SpscLinkedQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/util/UnsafeRefArrayAccess.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/package-info.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/maps/AbstractEntry.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscCompoundQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/BaseMpscLinkedAtomicArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/callback/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/AbstractArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/atomic/BaseMpscLinkedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/AbstractArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/AbstractArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/AbstractArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/AbstractArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1257262349_1642793356.2/0/jctools-core-2-1-1-sources-jar/org/jctools/queues/AbstractArrayQueue.java
No license file was found, but licenses were detected in source scan.

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>org.jctools</groupId>
  <artifactId>jctools-core</artifactId>
  <version>2.1.1</version>
  <name>Java Concurrency Tools Core Library</name>
  <description>Java Concurrency Tools Core Library</description>
  <packaging>bundle</packaging>

  <dependencies>
    <dependency>
      <groupId>org.hamcrest</groupId>
      <artifactId>hamcrest-all</artifactId>
      <version>${hamcrest.version}</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
<artifactId>maven-javadoc-plugin</artifactId>
<version>3.0.0-M1</version>
<configuration>
  <additionalparam>-Xdoclint:none</additionalparam>
</configuration>
<executions>
  <execution>
    <id>attach-javadocs</id>
    <goals>
      <goal>jar</goal>
    </goals>
  </execution>
</executions>
</plugin>
</plugins>
</build>
<distributionManagement>
  <repository>
    <id>bintray-jctools-jctools</id>
    <name>jctools-jctools</name>
    <url>https://api.bintray.com/maven/jctools/jctools/jctools-core/;publish=1</url>
  </repository>
</distributionManagement>
<url>https://github.com/JCTools</url>
<inceptionYear>2013</inceptionYear>
/licenses>
  <license>
    <name>Apache License, Version 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
    <distribution>repo</distribution>
  </license>
</licenses>
<scm>
  <url>https://github.com/JCTools/JCTools</url>
  <connection>scm:git:https://github.com/JCTools/JCTools</connection>
  <tag>HEAD</tag>
</scm>
<developers>
  <developer>
    <url>https://github.com/nitsanw</url>
  </developer>
  <developer>
    <url>https://github.com/mjpt777</url>
  </developer>
</developers>
1.1102 toidentifier 1.0.0 1.0.0

1.1102.1 Available under license:

MIT License

Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1103 jackson-jaxrs-base 2.10.4
1.1103.1 Available under license:
This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.1104 hk2-locator 2.6.1
1.1104.1 Available under license:
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license underLicensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and
b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) does not attempt to limit or alter the recipients’ rights in the Source Code under section 3.2; and
   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (“notices”) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.
Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.”

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE StATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR...
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks
Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.
1.1105 react-highlight-words 0.17.0 0.17.0

1.1105.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Treasure Data

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1106 babel-helper-replace-supers 6.24.1

6.24.1

1.1106.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.1107 is-extendable 0.1.1
1.1107.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1108 deepmerge 2.1.0 2.1.0
1.1108.1 Available under license:
The MIT License (MIT)

Copyright (c) 2012 Nicholas Fisher

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1109 @babel/plugin-syntax-object-rest-spread 7.8.3

1.1109.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1110 batik-ext 1.13

1.1110.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006 World Wide Web Consortium,
 * (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. This
* work is distributed under the W3C(r) Software License [1] in the hope that
* it will be useful, but WITHOUT ANY WARRANTY; without even the implied
* warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
*
*/

Found in path(s):
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/events/CustomEvent.java
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/events/KeyboardEvent.java
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/events/MutationNameEvent.java
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/events/TextEvent.java
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-jar/META-
INF/maven/org.apache.xmlgraphics/batik-ext/pom.xml
No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*

Found in path(s):
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/ElementTraversal.java
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/Location.java
* /opt/cola/permits/1146690638_1616599313.36/0/batik-ext-1-13-sources-1-
jar/org/apache/batik/w3c/dom/Window.java

1.1111 randexp 0.5.3 0.5.3
1.1111.1 Available under license:
MIT License

Copyright (C) 2011 by fent

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1112 hive-upgrade-acid 3.1.2
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Hive Upgrade Acid
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1113 which-module 2.0.0 2.0.0
1.1113.1 Available under license :
Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

1.1114 avro 1.9.1
1.1114.1 Available under license :
Apache Avro
Copyright 2010-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions.
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1115 hk2-api 2.2.0-b21

1.1115.1 Available under license :
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is
originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions
thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or
alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this
License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of
this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time.
Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the
license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of
the version of the License under which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under
any subsequent version of the License, You must distribute and make the Covered Software available under the
terms of the version of the License under which You originally received the Covered Software. Otherwise, You may
also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent
version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may
create and use a modified version of this License if You: (a) rename the license and remove any references to the
name of the license steward (except to note that the license differs from this License); and (b) otherwise make it
clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT
WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A
PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND
PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER
CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. section 252.227-7014(a)(1)) and "commercial
computer software documentation” as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
cat hk2-api-2.2.0-b21-sources/META-INF/inhabitants/default
#
# DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
#
# Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved.
#
# The contents of this file are subject to the terms of either the GNU
# General Public License Version 2 only ("GPL") or the Common Development
# and Distribution License("CDDL") (collectively, the "License"). You
# may not use this file except in compliance with the License. You can
# obtain a copy of the License at
# https://glassfish.dev.java.net/public/CDDL+GPL_1_1.html
# or packager/legal/LICENSE.txt. See the License for the specific
As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from
or based on this library. If you modify this library, you may extend
this exception to your version of the library, but you are not
obligated to do so. If you do not wish to do so, delete this
exception statement from your version.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software
10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.
1.1116 fast-deep-equal 1.1.0 1.1.0

1.1116.1 Available under license:
MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1117 atk 2.18.0 1

1.1117.1 Available under license:
GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.
This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is
The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.
The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a
table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility
is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or
table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.
This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or
link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

d) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the source code distributed need not include anything that is normally
distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

This package was debianized by Akira TAGOH <tagoh@debian.org> on Sun, 10 Mar 2002 20:41:39 +0900.

It was downloaded from <http://ftp.acc.umu.se/pub/GNOME/sources/atk/>.

Upstream Authors:
Bill.Haneman <bill.haneman@sun.com>
Marc.Mulcahy <marc.mulcahy@sun.com>
Padraig.Obriain <padraig.obriain@sun.com>

Copyright:

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the
Free Software Foundation, Inc., 51 Franklin St, Fifth Floor,
Boston, MA 02110-1301, USA.

See /usr/share/common-licenses/LGPL-2 on your debian system.

1.1118 @babel/template 7.10.4
1.1118.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1119 nan 2.14.2 2.14.2
1.1119.1 Available under license:
The MIT License (MIT)

==================================
Copyright (c) 2018 NAN contributors
-----------------------------------

*NAN contributors listed at <https://github.com/nodejs/nan#contributors>*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1120 tweetnacl 0.14.5 0.14.5
1.1120.1 Available under license :
This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or
distribute this software, either in source code form or as a compiled
binary, for any purpose, commercial or non-commercial, and by any
means.

In jurisdictions that recognize copyright laws, the author or authors
of this software dedicate any and all copyright interest in the
software to the public domain. We make this dedication for the benefit
of the public at large and to the detriment of our heirs and
successors. We intend this dedication to be an overt act of
relinquishment in perpetuity of all present and future rights to this
software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
 ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org>
**1.1121 json5 1.0.1 1.0.1**

**1.1121.1 Available under license:**

MIT License

Copyright (c) 2012-2018 Aseem Kishore, and [others].

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[others]: https://github.com/json5/json5/contributors

**1.1122 define-property 2.0.2 2.0.2**

**1.1122.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2015-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.1123 has-binary 0.1.7 0.1.7
1.1123.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Kevin Roark

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1124 lie 3.3.0 3.3.0
1.1124.1 Available under license:
#Copyright (c) 2014-2018 Calvin Metcalf, Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
1.1125 cglib 3.2.4

1.1125.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.1126 jersey-media-jaxb 2.25.1
1.1126.1 Available under license :

CDDL+GPL License

http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1127 bson 1.1.6 1.1.6
1.1127.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1128 jackson-annotations 2.10.5
1.1128.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and
   You must cause any modified files to carry prominent notices stating that You changed the files; and
   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1129 hadoop-mapreduce-client-common

3.2.2

1.1129.1 Available under license :

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-
2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4, which has the following notices:

The Netty Project
=================

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/
This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/
This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasternl.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
  * HOMEPAGE:
    * http://www.jdom.org/

The binary distribution of this product bundles binaries of
Commons Codec 1.4,
which has the following notices:
* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.
Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of
Commons Lang 2.6,
which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of
Apache Log4j 1.2.17,
which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
  Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
  TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
Jetty Web Container
Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
 ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/ (Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUnite Library Exception"

== Contributors ==
* Tatu Saloranta
* Providing benchmark suite
* Alec Wysoker
* Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:

* =========================================================================
  ==  NOTICE file corresponding to section 4(d) of the Apache License, ==
  ==  Version 2.0, in this case for the Apache Xerces Java distribution. ==
  =========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

The binary distribution of this product bundles binaries of Logback Classic Module 1.1.2, Logback Core Module 1.1.2, which has the following notices:

* Logback: the reliable, generic, fast and flexible logging framework. Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of Apache HBase - Annotations 1.2.6, Apache HBase - Client 1.2.6, Apache HBase - Common 1.2.6, Apache HBase - Hadoop Compatibility 1.2.6, Apache HBase - Hadoop Two Compatibility 1.2.6, Apache HBase - Prefix Tree 1.2.6, Apache HBase - Procedure 1.2.6, Apache HBase - Protocol 1.2.6, Apache HBase - Server 1.2.6, which has the following notices:

* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0,
which has the following notices:
  Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation
This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* ==================================================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  ==================================================================================

This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.
The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract
For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/*++
 *
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 * *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 * *
 * Abstract: The main routine
 *
--*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/\{lz4.h,lz4.c,lz4hc.h,lz4hc.c\},

/*
 LZ4 - Fast LZ compression algorithm
 Header File
 Copyright (C) 2011-2014, Yann Collet.
 BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

 Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

 * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

 THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

/*

For hadoop-common-project/hadoop-common/src/main/native/gtest
---------------------------------------------------------------------

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
re2j 1.1

---------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its contributors
  may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------------------------------------------------------------

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni
(https://github.com/fusesource/leveldbjni), which is available under the
following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of FuseSource Corp. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-
nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:
Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from
the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

=============

The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following
license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy
(http://code.google.com/p/snappy/), which is available under the following
license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
--------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js
--------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

---

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
  hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
  hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
  hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
  bootstrap v3.3.6
  broccoli-asset-rev v2.4.2
  broccoli-funnel v1.0.1
  datatables v1.10.8
  em-helper v0.5.13
  em-table v0.1.6
  ember v2.2.0
  ember-array-contains-helper v1.0.2
  ember-bootstrap v0.5.1
  ember-cli v1.13.13
  ember-cli-app-version v1.0.0
  ember-cli-babel v5.1.6
  ember-cli-content-security-policy v0.4.0
  ember-cli-dependency-checker v1.2.0
  ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodeings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

-----------------------------------------------------------------------------------------------

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------------------------------

For:
.hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js

-----------------------------------------------------------------------------------------------

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY.
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jar

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

For:
oj! Algorithms - version 43.0

-----------------------------------------------------------------------------------------------


This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as
documented below:

====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
  hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
  hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
  HSQLDB Database 2.3.4
COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group.”

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other...
than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.
2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are
effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms...
You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY
PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY
CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF
GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER
COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE
POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO
LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO
THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT
ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO
THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in
48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as
that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer
software documentation as such terms are used in 48C.F.R.12.212 (Sept.
227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software
with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement
concerning subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to
make it enforceable. This License shall be governed by the law of the
jurisdiction specified in a notice contained within the Original Software
(except to the extent applicable law, if any, provides otherwise), excluding
such jurisdictions conflict-of-law provisions. Any litigation relating to this
License shall be subject to the jurisdiction of the courts located in the
jurisdiction and venue specified in a notice contained within the Original
Software, with the losing party responsible for costs, including, without
limitation, court costs and reasonable attorneys fees and expenses. The
application of the United Nations Convention on Contracts for the International
Sale of Goods is expressly excluded. Any law or regulation which provides that
the language of a contract shall be construed against the drafter shall not
apply to this License. You agree that You alone are responsible for compliance
with the United States export administration regulations (and the export control
laws and regulation of any other countries) when You use, distribute or
otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is
responsible for claims and damages arising, directly or indirectly, out of its
utilization of rights under this License and You agree to work with Initial
Developer and Contributors to distribute such responsibility on an equitable
basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
gizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of
the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that
identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed
or otherwise made available under any subsequent version of the License, You
must distribute and make the Covered Software available under the terms of the
version of the License under which You originally received the Covered Software.
Otherwise, You may also choose to use, distribute or otherwise make the Covered
Software available under the terms of any subsequent version of the License
published by the license steward.
4.3. Modified Versions.
When You are an
Initial Developer and You want to create a new license for Your Original
Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward
(except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS
LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED
SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY
RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF
WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED
SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will
terminate automatically if You fail to comply with terms herein and fail to cure
such breach within 30 days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination of this License shall
survive.
6.2. If You assert a patent infringement claim (excluding declaratory
judgment actions) against Initial Developer or a Contributor (the Initial
Developer or Contributor against whom You assert such claim is referred to as
Participant) alleging that the Participant Software (meaning the
Contributor Version where the Participant is a Contributor or the Original
Software where the Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or indirectly to
You by such Participant, the Initial Developer (if the Initial Developer is not
the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this
License shall, upon 60 days notice from Participant terminate prospectively and
automatically at the expiration of such 60 day notice period, unless if within
such 60 day period You withdraw Your claim with respect to the Participant
Software against such Participant either unilaterally or pursuant to a written
agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to
make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:

Protocol Buffer Java API 2.5.0

--------------------------------------------------------------------------------
This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04
--------------------------------------------------------------------------------
By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation,
with or without modification, for any purpose and without fee or royalty is
hereby granted, provided that you include the following on ALL copies of the
software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the
  redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and
  conditions. If none exist, the W3C Software Short Notice should be included
  (hypertext is preferred, text is permitted) within the body of any redistributed
  or derivative code.
- Notice of any changes or modifications to the files, including the date changes
  were made. (We recommend you provide URIs to the location from which the code is
  derived.)

The binary distribution of this product bundles these dependencies under the
following license:
JUnit 4.11
Eclipse JDT Core 3.1.1
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form.

This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the
following license:
"Java Concurrency in Practice" book annotations 1.0
--------------------------------------------------------------------------------
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS
PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR
OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE
BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR/grants you the rights contained
here in consideration of your acceptance of such terms and conditions.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or
encyclopedia, in which the Work in its entirety in unmodified form, along with a
number of other contributions, constituting separate and independent works in
themselves, are assembled into a collective whole. A work that constitutes a
Collective Work will not be considered a Derivative Work (as defined below) for
the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other
pre-existing works, such as a translation, musical arrangement, dramatization,
fictionalization, motion picture version, sound recording, art reproduction,
abridgment, condensation, or any other form in which the Work may be recast,
transformed, or adapted, except that a work that constitutes a Collective Work
will not be considered a Derivative Work for the purpose of this License. For
the avoidance of doubt, where the Work is a musical composition or sound
recording, the synchronization of the Work in timed-relation with a moving image
("synching") will be considered a Derivative Work for the purpose of this
License.

"Licensor" means the individual or entity that offers the Work under the terms
of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of
this License.

"You" means an individual or entity exercising rights under this License who has
not previously violated the terms of this License with respect to the Work, or
who has received express permission from the Licensor to exercise rights under
this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or
restrict any rights arising from fair use, first sale or other limitations on
the exclusive rights of the copyright owner under copyright law or other
applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor
hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the
duration of the applicable copyright) license to exercise the rights in the Work
as stated below:

to reproduce the Work, to incorporate the Work into one or more Collective
Works, and to reproduce the Work as incorporated in the Collective Works;
to create and reproduce Derivative Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission the Work including as
incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive
inght to collect, whether individually or via a performance rights society (e.g.
ASCAP, BMI, SESAC), royalties for the public performance or public digital
performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right
to collect, whether individually or via a music rights agency or designated
agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created
by 17 USC Section 115 of the US Copyright Act (or the equivalent in other
jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the
Work is a sound recording, Licensor waives the exclusive right to collect,
whether individually or via a performance-rights society (e.g. SoundExchange),
royalties for the public digital performance (e.g. webcast) of the Work, subject
to the compulsory license created by 17 USC Section 114 of the US Copyright Act
The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer
UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

```
jamon-runtime 2.4.1
```

(MPL 2.0)

Mozilla Public License

Version 2.0

1. Definitions

1.1. Contributor

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributors Contribution.

1.3. Contribution

means Covered Software of a particular Contributor.

1.4. Covered Software

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses

means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form

means any form of the work other than Source Code Form.

1.7. Larger Work

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.
1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the
time of the initial grant or subsequently, any and all of the rights conveyed by
this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or
modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and
apparatus claims, in any patent Licensable by such Contributor that would be
infringed, but for the grant of the License, by the making, using, selling,
offering for sale, having made, import, or transfer of either its Contributions
or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General
Public License, Version 2.1, the GNU Affero General Public License, Version 3.0,
or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For
legal entities, You includes any entity that controls, is controlled by,
or is under common control with You. For purposes of this definition,
control means (a) the power, direct or indirect, to cause the direction or
management of such entity, whether by contract or otherwise, or (b) ownership of
more than fifty percent (50%) of the outstanding shares or beneficial ownership
of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:
under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.
2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).
3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and
cross-claims) alleging that a Contributor Version directly or indirectly
infringes any patent, then the rights granted to You by any and all Contributors
for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user
license agreements (excluding distributors and resellers) which have been
validly granted by You or Your distributors under this License prior to
termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without
warranty of any kind, either expressed, implied, or statutory, including,
without limitation, warranties that the Covered Software is free of defects,
merchantable, fit for a particular purpose or non-infringing. The entire risk as
to the quality and performance of the Covered Software is with You. Should any
Covered Software prove defective in any respect, You (not any Contributor)
assume the cost of any necessary servicing, repair, or correction. This
disclaimer of warranty constitutes an essential part of this License. No use of
any Covered Software is authorized under this License except under this
disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including
negligence), contract, or otherwise, shall any Contributor, or anyone who
distributes Covered Software as permitted above, be liable to You for any
direct, indirect, special, incidental, or consequential damages of any character
including, without limitation, damages for lost profits, loss of goodwill, work
stoppage, computer failure or malfunction, or any and all other commercial
damages or losses, even if such party shall have been informed of the
possibility of such damages. This limitation of liability shall not apply to
liability for death or personal injury resulting from such party’s negligence
to the extent applicable law prohibits such limitation. Some jurisdictions do
not allow the exclusion or limitation of incidental or consequential damages, so
this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a
jurisdiction where the defendant maintains its principal place of business and
such litigation shall be governed by laws of that jurisdiction, without
reference to its conflict-of-law provisions. Nothing in this Section shall
prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter
hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice
This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

--------------------------------------------------------------------------------
/*--
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:
   “This product includes software developed by the JDOM Project (http://www.jdom.org/).”
Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the JDOM Project and was originally
created by Jason Hunter <jhunter_AT_jdom_DOT_org> and
Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information
on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the
following license:
Hbase Server 1.2.4

--------------------------------------------------------------------------------

This project bundles a derivative image for our Orca Logo. This image is
available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN
ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES
REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR
DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE
COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY
COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS
AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE
TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY
BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS
CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND
CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and
other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works
expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";

c. to Distribute and Publicly Perform the Work including as incorporated
in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an
Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URL, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations).
5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made
available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

--------------------------------------------------------------------------------

For: hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
    /server/datanode/checker/AbstractFuture.java and
hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs
    /server/datanode/checker/TimeoutFuture.java

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1130 @babel/helper-validator-identifier

7.15.7

1.1130.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
1.1131 drill-jdbc 1.19.0
1.1131.1 Available under license :

Drill : Exec : JDBC Driver using dependencies
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1132 dashdash 1.14.1 1.14.1
1.1132.1 Available under license :

# This is the MIT license

Copyright (c) 2013 Trent Mick. All rights reserved.
Copyright (c) 2013 Joyent Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1133 babel-preset-stage-2 6.24.1
1.1133.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
Copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1134 object.pick 1.3.0 1.3.0

1.1134.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1135 concat-map 0.0.1 0.0.1
1.1135.1 Available under license :
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1136 ws 6.2.1 6.2.1
1.1136.1 Available under license :
The MIT License (MIT)

Copyright (c) 2011 Einar Otto Stangvik <einaros@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1137 nyc 15.1.0
1.1137.1 Available under license :
   ISC License

   Copyright (c) 2015, Contributors

   Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

   THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1138 klaw 1.3.1 1.3.1
1.1138.1 Available under license :
   (The MIT License)

   Copyright (c) 2015-2016 JP Richardson

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files
(the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy,
modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1139 fs.realpath 1.0.0

1.1139.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----

This library bundles a version of the `fs.realpath` and `fs.realpathSync` methods from Node.js v0.10 under the terms of the Node.js MIT license.

Node's license follows, also included at the header of `old.js` which contains the licensed code:

Copyright Joyent, Inc. and other Node contributors.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1140 etag 1.8.1 1.8.1
1.1140.1 Available under license :
(The MIT License)

Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
**1.1141 esprima 2.7.3**

**1.1141.1 Available under license:**

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**1.1142 jsonpointer 4.0.1 4.0.1**

**1.1142.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2011-2015 Jan Lehnardt <jan@apache.org> & Marc Bachmann <https://github.com/marcbachmann>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1143 bunyan 1.8.1 1.8.1

1.1143.1 Available under license:

# This is the MIT license

Copyright 2016 Trent Mick
Copyright 2016 Joyent Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1144 @babel/plugin-transform-object-super 7.16.0

1.1144.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1145 path-to-regexp 2.1.0 2.1.0

1.1145.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.


1.1146.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1147 jnr-ffi 2.1.5

1.1147.1 Available under license :

No license file was found, but licenses were detected in source scan.

    /*
    * Copyright (C) 2013 Wayne Meissner
    *
    * This file is part of the JNR project.
    *
    * Licensed under the Apache License, Version 2.0 (the "License");
    * you may not use this file except in compliance with the License.
    * You may obtain a copy of the License at
    *
    *    http://www.apache.org/licenses/LICENSE-2.0
    *
    * Unless required by applicable law or agreed to in writing, software
    * distributed under the License is distributed on an "AS IS" BASIS,
    * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
    * See the License for the specific language governing permissions and
    * limitations under the License.
    */

    Found in path(s):
    * /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/InterfaceScanner.java
    * /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/FromNativeType.java
    * /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/mapper/SimpleFunctionMapper.java
    * /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/StructByReferenceResultConverterFactory.java
    * /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-
Open Source Used In DNAC Platform Arica 4123
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/FFIProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010 The Miyamoto Team
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/AnnotationProxy.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/AnnotationProperty.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/
Open Source Used In DNAC Platform Arica  4125
/*
 * Copyright (C) 2010 Wayne Meissner
 * 
 * This file is part of the JNR project.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *    http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/StructByReferenceFromNativeConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/jffi/NullObjectParameterStrategy.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/NativeLong32ArrayParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/BoxedFloatArrayParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/jffi/NoX86.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/ClosureFromNativeConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/StructArrayParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/CharSequenceArrayParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/platform/sparcv9/solaris/TypeAliases.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/StringBufferParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/jffi/PointerParameterStrategy.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/types/u_int16_t.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/ByReferenceParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/provider/converters/BoxedFloatArrayParameterConverter.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources.jar/jnr/ffi/types/gid_t.java

Open Source Used In DNAC Platform Arica  4129
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
 * Copyright (C) 2008-2012 Wayne Meissner
 *
 * This file is part of the JNR project.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/DefaultInvokerFactory.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/NativeInvocationHandler.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/Pointer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2010 Wayne Meissner
 * 
 * This file is part of the JNR project.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/NativeType.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008-2010 Wayne Meissner
 * 
 * This file is part of the JNR project.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Open Source Used in DNAC Platform Arica
jar/jnr/ffi/provider/jffi/ArrayMemoryIO.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/annotations/In.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/ByteBufferMemoryIO.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/AsmRuntime.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/CodegenUtils.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/MemoryManager.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/AbstractX86StubCompiler.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/NativeLong.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/byref/ShortByReference.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/byref/ByReference.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/internal/Finalizer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
* Copyright (C) 2008-2010 Wayne Meissner
* 
* This file is part of the JNR project.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/* Some of the design and code of this class is from the javolution project.
* 
* Copyright (C) 2006 - Javolution (http://javolution.org/)
* All rights reserved.
* 
* Permission to use, copy, modify, and distribute this software is
* freely granted, provided that this notice is preserved.
*/

/*
* Copyright (C) 2011 Wayne Meissner
*/

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/FinalizablePhantomReference.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/FinalizableSoftReference.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/FinalizableReferenceQueue.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/FinalizableWeakReference.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/util/ref/FinalizableReference.java

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/Struct.java
* /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/Union.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 Wayne Meissner
*/
This file is part of the JNR project.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/BaseMethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/NativeClosurePointer.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/ObjectReferenceManager.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/IntPointer.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/NativeClosureFactory.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/FastNumericMethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/AsmBuilder.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/ClosureManager.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/FastLongMethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/annotations/Delegate.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/NativeClosureManager.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/StructLayout.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/X86MethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/AbstractAsmLibraryInterface.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/annotations/LongLong.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/DefaultObjectReferenceManager.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/MethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/ClosureUtil.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-jar/jnr/ffi/provider/jffi/BufferMethodGenerator.java
- /opt/cola/permits/1258272141_1643050049.11/0/jnr-ffi-2-1-5-sources-
1.1148 drill-druid-storage 1.19.0

1.1148.1 Available under license:

Drill : Contrib : Storage : Druid
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1149 cli-cursor 2.1.0 2.1.0
1.1149.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1150 string-prototype-codepointat 0.2.1

0.2.1

1.1150.1 Available under license :

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1151 is-fullwidth-code-point 2.0.0 2.0.0

1.1151.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1152 schema-utils 0.4.7 0.4.7

1.1152.1 Available under license :

Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1153 zjsonpatch 0.3.6

1.1153.1 Available under license :

Apache License

Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1154 pip 9.0.1
1.1154.1 Available under license :
Copyright (c) 2008-2016 The pip developers (see AUTHORS.txt file)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1155 babel-plugin-transform-es2015-literals
6.22.0 6.22.0
1.1155.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1156 eclipse-egit-github-core 2.1.5
1.1156.1 Available under license:

/* Copyright (c) 2011 GitHub Inc.
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * which accompanies this distribution, and is available at
 *
 * Contributors:
 * Kevin Sawicki (GitHub Inc.) - initial API and implementation
 */
package org.eclipse.egit.github.core;

import java.io.Serializable;

/**
 * Repository contributor model class
 */
public class Contributor implements Serializable {

/** serialVersionUID */
private static final long serialVersionUID = -8434028880839230626L;

/**
 * Anonymous contributor type value
 */
public static final String TYPE_ANONYMOUS = "Anonymous"; //SNON-NLS-1$

private int contributions;

private int id;

private String avatarUrl;

private String login;

private String name;

private String type;

private String url;
/**
 * @return contributions
 */
public int getContributions() {
    return contributions;
}

/**
 * @param contributions
 * @return this contributor
 */
public Contributor setContributions(int contributions) {
    this.contributions = contributions;
    return this;
}

/**
 * @return id
 */
public int getId() {
    return id;
}

/**
 * @param id
 * @return this contributor
 */
public Contributor setId(int id) {
    this.id = id;
    return this;
}

/**
 * @return avatarUrl
 */
public String getAvatarUrl() {
    return avatarUrl;
}

/**
 * @param avatarUrl
 * @return this contributor
 */
public Contributor setAvatarUrl(String avatarUrl) {
    this.avatarUrl = avatarUrl;
    return this;
}
/**
 * @return login
 */
public String getLogin() {
  return login;
}

/**
 * @param login
 * @return this contributor
 */
public Contributor setLogin(String login) {
  this.login = login;
  return this;
}

/**
 * @return name
 */
public String getName() {
  return name;
}

/**
 * @param name
 * @return this contributor
 */
public Contributor setName(String name) {
  this.name = name;
  return this;
}

/**
 * @return type
 */
public String getType() {
  return type;
}

/**
 * @param type
 * @return this contributor
 */
public Contributor setType(String type) {
  this.type = type;
  return this;
}
/**
 * @return url
 */
public String getUrl() {
    return url;
}

/**
 * @param url
 * @return this contributor
 */
public Contributor setUrl(String url) {
    this.url = url;
    return this;
}

1.1157 jwa 1.4.1 1.4.1

1.1157.1 Available under license:
Copyright (c) 2013 Brian J. Brennan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1158 minlog 1.3.0
1.1159 commons-lang3 3.8.1

1.1159.1 Available under license:

Apache Commons Lang

Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

Open Source Used In DNAC Platform Arica 4153
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1160 npm-run-path 2.0.2 2.0.2

1.1160.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1161 babel-plugin-transform-es2015-destructuring 6.23.0 6.23.0

1.1161.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1162 binary 0.3.0 0.3.0

1.1162.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1163 avatica-core 1.17.0

1.1163.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Calcite -- Avatica
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1164 groovy 3.0.8
Open Source Used In DNAC Platform Arica 4163

1.1164.1 Available under license:
No license file was found, but licenses were detected in source scan.

/** Java 1.5 Recognizer
 *
 * Run 'java Main [-showtree] directory-full-of-java-files'
 *
 * [The -showtree option pops up a Swing frame that shows
 * the AST constructed from the parser.]
 *
 * Run 'java Main <directory full of java files>'
 *
 * Contributing authors:
 * Jeremy Rayner groovy@ross-rayner.com
 * John Mitchell johnm@non.net
 * Terence Parr partr@magelang.com
 * John Lilley jilley@empathy.com
 * Scott Stanchfield thetick@magelang.com
 * Markus Mohnen mohnen@informatik.rwth-aachen.de
 * Peter Williams pete.williams@sun.com
 * Allan Jacobs Allan.Jacobs@eng.sun.com
 * Steve Messick messick@redhills.com
 * John Pybus john@pybus.org
 *
 * Version 1.00 December 9, 1997 -- initial release
 * Version 1.01 December 10, 1997
 * fixed bug in octal def (0..7 not 0..8)
 * Version 1.10 August 1998 (partr)
 * added tree construction
 * fixed definition of WS, comments for mac,pc,unix newlines
 * added unary plus
 * Version 1.11 (Nov 20, 1998)
 * Added "shutup" option to turn off last ambig warning.
 * Fixed inner class def to allow named class defs as statements
 * synchronized requires compound not simple statement
 * add [] after builtInType DOT class in primaryExpression
 * "const" is reserved but not valid..removed from modifiers
 * Version 1.12 (Feb 2, 1999)
 * Changed LITERAL_xxx to xxx in tree grammar.
 * Updated java.g to use tokens {...} now for 2.6.0 (new feature).
 *
 * Version 1.13 (Apr 23, 1999)
 * Didn't have (stat)? for else clause in tree parser.
 * Didn't gen ASTs for interface extends. Updated tree parser too.
 * Updated to 2.6.0.
 * Version 1.14 (Jun 20, 1999)
 * Allowed final/abstract on local classes.
 * Removed local interfaces from methods
*Put instanceof precedence where it belongs...in relationalExpr
*It also had expr not type as arg; fixed it.
*Missing ! on SEMI in classBlock
*fixed: (expr) + "string" was parsed incorrectly (+ as unary plus).
*fixed: didn't like Object[].class in parser or tree parser
* Version 1.15 (Jun 26, 1999)
*Screwed up rule with instanceof in it. :( Fixed.
*Tree parser didn't like (expr).something; fixed.
*Allowed multiple inheritance in tree grammar. oops.
* Version 1.16 (August 22, 1999)
*Extending an interface built a wacky tree: had extra EXTENDS.
*Tree grammar didn't allow multiple superinterfaces.
*Tree grammar didn't allow empty var initializer: {}
* Version 1.17 (October 12, 1999)
*ESC lexer rule allowed 399 max not 377 max.
*java.tree.g didn't handle the expression of synchronized
*statements.
* Version 1.18 (August 12, 2001)
* Terence updated to Java 2 Version 1.3 by
*observing/combining work of Allan Jacobs and Steve
*Messick. Handles 1.3 src. Summary:
*o primary didn't include boolean.class kind of thing
* o constructor calls parsed explicitly now:
* see explicitConstructorInvocation
*o add strictfp modifier
* o missing objBlock after new expression in tree grammar
*o merged local class definition alternatives, moved after declaration
*o fixed problem with ClassName.super.field
* o reordered some alternatives to make things more efficient
*o long and double constants were not differentiated from int/float
*o whitespace rule was inefficient: matched only one char
*o add an examples directory with some nasty 1.3 cases
*o made Main.java use buffered IO and a Reader for Unicode support
*o supports UNICODE?
* Using Unicode charVocabulary makes code file big, but only
* in the bitsets at the end. I need to make ANTLR generate
* unicode bitsets more efficiently.
* Version 1.19 (April 25, 2002)
*Terence added in nice fixes by John Pybus concerning floating
*constants and problems with super() calls. John did a nice
*reorg of the primary/postfix expression stuff to read better
*and makes f.g.super() parse properly (it was METHOD_CALL not
*a SUPER_CTOR_CALL). Also:
*
*o "finally" clause was a root...made it a child of "try"
*o Added stuff for asserts too for Java 1.4, but *commented out*
* as it is not backward compatible.
* Version 1.20 (October 27, 2002)
* 
* Terence ended up reorging John Pybus' stuff to
* remove some nondeterminisms and some syntactic predicates.
* Note that the grammar is stricter now; e.g., this(...) must
* be the first statement.
* 
* Ternary ?: operator wasn't working as array name:
* (isBig ? bigDigits : digits)[i]; 
* 
* Checked parser/tree parser on source for
* Resin-2.0.5, jive-2.1.1, jdk 1.3.1, Lucene, antlr 2.7.2a4,
* and the 110k-line jGuru server source.
* 
* Version 1.21 (October 17, 2003)
* Fixed lots of problems including:
* Ray Waldin: add typeDefinition to interfaceBlock in java.tree.g
* He found a problem/fix with floating point that start with 0
* Ray also fixed problem that (int.class) was not recognized.
* Thorsten van Ellen noticed that \ are allowed incorrectly in strings.
* TJP fixed CHAR_LITERAL analogously.
* 
* Changes by Matt Quail to support generics (as per JDK1.5/JSR14)
* Notes:
* o We only allow the "extends" keyword and not the "implements"
* keyword, since that's what JSR14 seems to imply.
* o Thanks to Monty Zukowski for his help on the antlr-interest
* mail list.
* o Thanks to Alan Eliasen for testing the grammar over his
* Fink source base
* 
* Version 1.22 (July, 2004)
* Changes by Michael Studman to support Java 1.5 language extensions
* Notes:
* o Added support for annotations types
* o Finished off Matt Quail's generics enhancements to support bound type arguments
* o Added support for new for statement syntax
* o Added support for static import syntax
* o Added support for enum types
* o Tested against JDK 1.5 source base and source base of jdigraph project
* o Thanks to Matt Quail for doing the hard part by doing most of the generics work
* 
* Version 1.22.1 (July 28, 2004)
* Bug omission fixes for Java 1.5 language support
* o Fixed tree structure bug with classOrInterface - thanks to Pieter Vangorpto for
* spotting this
* o Fixed bug where incorrect handling of SR and BSR tokens would cause type
*parameters to be recognised as type arguments.
*  o Enabled type parameters on constructors, annotations on enum constants
*  and package definitions
*  o Fixed problems when parsing if ((char.class.equals(c))) {} - solution by Matt Quail at Cenqua
*
* Version 1.22.2 (July 28, 2004)
*  Slight refactoring of Java 1.5 language support
*  o Refactored for/"foreach" productions so that original literal "for" literal
*  is still used but the for sub-clauses vary by token type
*  o Fixed bug where type parameter was not included in generic constructor's branch of AST
*
* Version 1.22.3 (August 26, 2004)
*  Bug fixes as identified by Michael Stahl; clean up of tabs/spaces
*       and other refactorings
*  o Fixed typeParameters omission in identPrimary and newStatement
*  o Replaced GT reconciliation code with simple semantic predicate
*  o Adapted enum/assert keyword checking support from Michael Stahl's java15 grammar
*  o Refactored typeDefinition production and field productions to reduce duplication
*
* Version 1.22.4 (October 21, 2004)
*  Small bux fixes
*  o Added typeArguments to explicitConstructorInvocation, e.g. new <String>MyParameterised()
*  o Added typeArguments to postfixExpression productions for anonymous inner class super
*     constructor invocation, e.g. new Outer().<String>super()
*  o Fixed bug in array declarations identified by Geoff Roy
*
* Version 1.22.4.j.1
*  Changes by Jeremy Rayner to support java2groovy tool
*  o I have taken java.g for Java1.5 from Michael Studman (1.22.4)
*  and have made some changes to enable use by java2groovy tool (Jan 2007)
*
* Based on an original grammar released in the PUBLIC DOMAIN
*/

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/java/JavaRecognizer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 Google Inc. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/EvictionListener.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/Weighers.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/Weigher.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/ConcurrentLinkedHashMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 Google Inc. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/EntryWeigher.java
No license file was found, but licenses were detected in source scan.

/*
* This file is adapted from the Antlr4 Java grammar which has the following license
*
* Copyright (c) 2013 Terence Parr, Sam Harwell
* All rights reserved.
* [The "BSD licence"]
*
* http://www.opensource.org/licenses/bsd-license.php
*
* Subsequent modifications by the Groovy community have been done under the Apache License v2:
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

/**
 * The Groovy grammar is based on the official grammar for Java:
 * https://github.com/antlr/grammars-v4/blob/master/java/Java.g4
 */
lexer grammar GroovyLexer;

options {
    superClass = AbstractLexer;
}

@header {
    import java.util.Deque;
    import java.util.ArrayDeque;
    import java.util.Map;
    import java.util.HashMap;
    import java.util.Set;
    import java.util.HashSet;
    import java.util.Collections;
    import java.util.stream.IntStream;
    import java.util.logging.Logger;
    import java.util.logging.Level;
    import org.apache.groovy.util.Maps;
    import static org.apache.groovy.parser.antlr4.SemanticPredicates.*;
}

@members {
private static final Logger LOGGER = Logger.getLogger(GroovyLexer.class.getName());

private long tokenIndex = 0;
private int lastTokenType = 0;
private int invalidDigitCount = 0;

/**
 * Record the index and token type of the current token while emitting tokens.
 */
@Override
public void emit(Token token) {
    this.tokenIndex++;

    int tokenType = token.getType();
    if (Token.DEFAULT_CHANNEL == token.getChannel()) {
        this.lastTokenType = tokenType;
    }

    if (RollBackOne == tokenType) {
        this.rollbackOneChar();
    }

    super.emit(token);
}

private static final int[] REGEX_CHECK_ARRAY =
    IntStream.of(
        Identifier, CapitalizedIdentifier, NullLiteral, BooleanLiteral, THIS, RPAREN, RBRACK,
        RBRACE,
        IntegerLiteral, FloatingPointLiteral, StringLiteral, GStringEnd, INC, DEC
    ).sorted().toArray();

private boolean isRegexAllowed() {
    if (Arrays.binarySearch(REGEX_CHECK_ARRAY, this.lastTokenType) >= 0) {
        return false;
    }
    return true;
}

/**
 * just a hook, which will be overrided by GroovyLangLexer
 */
protected void rollbackOneChar() {}

private static class Paren {
    private String text;
    private int lastTokenType;
    private int line;

    /*
     *
private int column;

public Paren(String text, int lastTokenType, int line, int column) {
    this.text = text;
    this.lastTokenType = lastTokenType;
    this.line = line;
    this.column = column;
}

public String getText() {
    return this.text;
}

public int getLastTokenType() {
    return this.lastTokenType;
}

public int getLine() {
    return line;
}

public int getColumn() {
    return column;
}

@Override
public int hashCode() {
    return (int) (text.hashCode() * line + column);
}

@Override
public boolean equals(Object obj) {
    if (!(obj instanceof Paren)) {
        return false;
    }
    Paren other = (Paren) obj;
    return this.text.equals(other.text) && (this.line == other.line && this.column == other.column);
}

protected void enterParenCallback(String text) {}

protected void exitParenCallback(String text) {}

private final Deque<Paren> parenStack = new ArrayDeque<>(32);
private void enterParen() {
    String text = getText();
    enterParenCallback(text);
}

parentStack.push(new Paren(text, this.lastTokenType, getLine(), getCharPositionInLine()));
}

private void exitParen() {
    String text = getText();
    exitParenCallback(text);
    Paren paren = parenStack.peek();
    if (null == paren) return;
    parenStack.pop();
}

private boolean isInsideParens() {
    Paren paren = parenStack.peek();
    // We just care about "(" and "]", inside which the new lines will be ignored.
    // Notice: the new lines between "{" and "}" can not be ignored.
    if (null == paren) {
        return false;
    } else {
        return ("(".equals(paren.getText()) && TRY != paren.getLastTokenType()) // we don't treat try-paren(i.e. try
        (....)) as parenthesis
            || "]".equals(paren.getText());
    }
}

private void ignoreTokenInsideParens() {
    if (!this.isInsideParens()) {
        return;
    }
    this.setChannel(Token.HIDDEN_CHANNEL);
}

private void ignoreMultiLineCommentConditionally() {
    if (!this.isInsideParens() && isFollowedByWhiteSpaces(_input)) {
        return;
    }
    this.setChannel(Token.HIDDEN_CHANNEL);
}

@Override
public int getSyntaxErrorSource() {
    return GroovySyntaxError.LEXER;
}
@Override
public int getErrorLine() {
    return getLine();
}

@Override
public int getErrorColumn() {
    return getCharPositionInLine() + 1;
}

@Override
public int popMode() {
    try {
        return super.popMode();
    } catch (EmptyStackException ignored) { // raised when parens are unmatched: too many ), ], or ]
        if (LOGGER.isLoggable(Level.FINEST)) {
            LOGGER.finest(org.codehaus.groovy.runtime.DefaultGroovyMethods.asString(ignored));
        }
    }

    return Integer.MIN_VALUE;
}

private static boolean isJavaIdentifierStartAndNotIdentifierIgnorable(int codePoint) {
    return Character.isJavaIdentifierStart(codePoint) && !Character.isIdentifierIgnorable(codePoint);
}

private static boolean isJavaIdentifierPartAndNotIdentifierIgnorable(int codePoint) {
    return Character.isJavaIdentifierPart(codePoint) && !Character.isIdentifierIgnorable(codePoint);
}

// 3.10.5 String Literals
StringLiteral
   : GStringQuotationMark DqStringCharacter* GStringQuotationMark
| SqStringQuotationMark SqStringCharacter* SqStringQuotationMark
| Slash      { this.isRegexAllowed() && _input.LA(1) != '*' }?
    SlashyStringCharacter+ Slash
| TdqStringQuotationMark TdqStringCharacter* TdqStringQuotationMark
| TsqStringQuotationMark TsqStringCharacter* TsqStringQuotationMark
| DollarSlashyGStringQuotationMarkBegin DollarSlashyStringCharacter+ DollarSlashyGStringQuotationMarkEnd
| ;

// Groovy gstring
GStringBegin
   : GStringQuotationMark DqStringCharacter* Dollar -> pushMode(DQ_GSTRING_MODE),
     pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
TdqGStringBegin
   : TdqStringQuotationMark TdqStringCharacter* Dollar -> type(GStringBegin),
     pushMode(TDQ_GSTRING_MODE), pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
SlashyGStringBegin
   : Slash [ this.isRegexAllowed() && _input.LA(1) != '*' ]? SlashyStringCharacter* Dollar { 
     isFollowedByJavaLetterInGString(_input) }? -> type(GStringBegin), pushMode(SLASHY_GSTRING_MODE),
     pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
DollarSlashyGStringBegin
   : DollarSlashyGStringQuotationMarkBegin DollarSlashyStringCharacter* Dollar { 
     isFollowedByJavaLetterInGString(_input) }? -> type(GStringBegin),
     pushMode(DOLLAR_SLASHY_GSTRING_MODE), pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
mode DQ_GSTRING_MODE;
GStringEnd
   : GStringQuotationMark    -> popMode
   ;
GStringPart
   : Dollar   -> pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
GStringCharacter
   : DqStringCharacter -> more
   ;
mode TDQ_GSTRING_MODE;
TdqGStringEnd
   : TdqStringQuotationMark    -> type(GStringEnd), popMode
   ;
TdqGStringPart
   : Dollar   -> type(GStringPart), pushMode(GSTRING_TYPE_SELECTOR_MODE)
   ;
TdqGStringCharacter
   : TdqStringCharacter -> more
   ;
mode SLASHY_GSTRING_MODE;
SlashyGStringEnd
   : Dollar? Slash  -> type(GStringEnd), popMode
   ;
SlashyGStringPart
   : Dollar { isFollowedByJavaLetterInGString(_input) }? -> type(GStringPart),
     pushMode(GSTRING_TYPE_SELECTOR_MODE)
SlashyGStringCharacter
: SlashyStringCharacter -> more

mode DOLLAR_SLASHY_GSTRING_MODE;
DollarSlashyGStringEnd
: DollarSlashyGStringQuotationMarkEnd -> type(GStringEnd), popMode
;
DollarSlashyGStringPart
: Dollar { isFollowedByJavaLetterInGString(_input) }? -> type(GStringPart), pushMode(GSTRING_TYPE_SELECTOR_MODE)
;
DollarSlashyGStringCharacter
: DollarSlashyStringCharacter -> more
;

mode GSTRING_TYPE_SELECTOR_MODE;
GStringLBrace
: '{' { this.enterParen(); } -> type(LBRACE), popMode, pushMode(DEFAULT_MODE)
;
GStringIdentifier
: IdentifierInGString -> type(Identifier), popMode, pushMode(GSTRING_PATH_MODE)
;

mode GSTRING_PATH_MODE;
GStringPathPart
: Dot IdentifierInGString
;
RollBackOne
: . { // a trick to handle GStrings followed by EOF properly
int readChar = _input.LA(-1);
if (EOF == _input.LA(1) && ('"' == readChar || '\' == readChar)) {
    setType(GStringEnd);
} else {
    setChannel(HIDDEN);
}
} -> popMode
;

mode DEFAULT_MODE;
// character in the double quotation string. e.g. "a"
fragment
DqStringCharacter
: ~["\n\r\$]
// the reserved type name of Java10
VAR : 'var';

// 3.9 Keywords
BuiltInPrimitiveType
   : BOOLEAN
   | CHAR
   | BYTE
   | SHORT
   | INT
   | LONG
   | FLOAT
   | DOUBLE

ABSTRACT : 'abstract';
ASSERT : 'assert';

fragment
BOOLEAN : 'boolean';

BREAK : 'break';

fragment
BYTE : 'byte';

CASE : 'case';
CATCH : 'catch';

fragment
CHAR : 'char';

CLASS : 'class';
CONST : 'const';
CONTINUE : 'continue';
DEFAULT : 'default';
DO : 'do';

fragment
DOUBLE : 'double';

ELSE : 'else';
ENUM : 'enum';
EXTENDS : 'extends';
FINAL : 'final';
FINALLY : 'finally';

fragment
FLOAT       : 'float';

FOR         : 'for';
IF           : 'if';
GOTO         : 'goto';
IMPLEMENTES : 'implements';
IMPORT      : 'import';
INSTANCEOF  : 'instanceof';

fragment

INT          : 'int';

INTERFACE    : 'interface';

fragment

LONG         : 'long';

NATIVE       : 'native';
NEW          : 'new';
PACKAGE      : 'package';
PRIVATE      : 'private';
PROTECTED    : 'protected';
PUBLIC       : 'public';
RETURN       : 'return';

fragment

SHORT        : 'short';

STATIC       : 'static';
STRUCTFP     : 'strictfp';
SUPER        : 'super';
SWITCH       : 'switch';
SYNCHRONIZED : 'synchronized';
THIS         : 'this';
THROW        : 'throw';
THROWS       : 'throws';
TRANSIENT    : 'transient';
TRY          : 'try';
VOID         : 'void';
VOLATILE     : 'volatile';
WHILE        : 'while';

// 3.10.1 Integer Literals

IntegerLiteral
Open Source Used In DNAC Platform Arica 4178
Digits
  :  Digit (DigitOrUnderscore* Digit)?

fragment
Digit
  :  Zero
      |  NonZeroDigit

fragment
NonZeroDigit
  :  [1-9]

fragment
DigitOrUnderscore
  :  Digit
      |  Underscore

fragment
Underscores
  :  Underscore+

fragment
Underscore
  :  '_'

fragment
HexNumeral
  :  Zero [xX] HexDigits

fragment
HexDigits
  :  HexDigit (HexDigitOrUnderscore* HexDigit)?

fragment
HexDigit
  :  [0-9a-fA-F]

fragment
HexDigitOrUnderscore
// 3.10.2 Floating-Point Literals
FloatingPointLiteral
: ( DecimalFloatingPointLiteral
  | HexadecimalFloatingPointLiteral
) (Underscore ( require(false, "Number ending with underscores is invalid", -1, true); ));

fragment
DecimalFloatingPointLiteral
: Digits Dot Digits ExponentPart? FloatTypeSuffix?
  | Digits ExponentPart FloatTypeSuffix?
  | Digits FloatTypeSuffix
  ;

fragment
ExponentPart
: ExponentIndicator SignedInteger
  ;

fragment
ExponentIndicator
: [eE]
  ;

fragment
SignedInteger
: Sign? Digits
  ;

fragment
Sign
: [+\-]
  ;

fragment
FloatTypeSuffix
: [fFdDgG]
  ;

fragment
HexadecimalFloatingPointLiteral
: HexSignificand BinaryExponent FloatTypeSuffix?
  ;

fragment
HexSignificand
: HexNumeral Dot?
  | Zero [xX] HexDigits? Dot HexDigits
  ;
fragment
  BinaryExponent
    :  BinaryExponentIndicator SignedInteger
    ;

fragment
  BinaryExponentIndicator
    :  [pP]
    ;

fragment
  Dot :  '.'
    ;

// 3.10.3 Boolean Literals

BooleanLiteral
  :  'true'
    |  'false'
    ;

// 3.10.6 Escape Sequences for Character and String Literals

fragment
  EscapeSequence
    :  Backslash [btnfrs"\]
    |  OctalEscape
    |  UnicodeEscape
    |  DollarEscape
    |  LineEscape
    ;

fragment
  OctalEscape
    :  Backslash OctalDigit
    |  Backslash OctalDigit OctalDigit
    |  Backslash ZeroToThree OctalDigit OctalDigit
    ;

// Groovy allows 1 or more u's after the backslash
fragment
  UnicodeEscape
    :  Backslash 'u' HexDigit HexDigit HexDigit HexDigit
    ;
fragment
ZeroToThree
  : [0-3]

// Groovy Escape Sequences

fragment
DollarEscape
  : Backslash Dollar

fragment
LineEscape
  : Backslash LineTerminator

fragment
LineTerminator
  : \n? \n | \r

fragment
SlashEscape
  : Backslash Slash

fragment
Backslash
  : \\n
fragment
Slash
  : '/'

fragment
Dollar
  : '$'

fragment
GStringQuotationMark
  : '''
SqStringQuotationMark
  : \" 
  ;

fragment
TdqStringQuotationMark
  : """
  ;

fragment
TsqStringQuotationMark
  : \"\"\"
  ;

fragment
DollarSlashyGStringQuotationMarkBegin
  : "/$"
  ;

fragment
DollarSlashyGStringQuotationMarkEnd
  : '/$'
  ;

fragment
DollarSlashEscape
  : '/$'
  ;

fragment
DollarDollarEscape
  : '/$'
  ;

// 3.10.7 The Null Literal
NullLiteral
  : 'null'
  ;

// Groovy Operators

RANGE_INCLUSIVE  : '..';
RANGE_EXCLUSIVE  : '..<';
SPREAD_DOT       : '*.';
SAFE_DOT         : '?.';
SAFE_CHAIN_DOT   : '???';
ELVIS            : '?:';
METHOD_POINTER   : '.&';
METHOD_REFERENCE     : '::';
REGEX_FIND           : '~=';
REGEX_MATCH          : '==~';
POWER                : '***';
POWER_ASSIGN         : '* *=';
SPACESHIP            : '<=>';
IDENTICAL            : '===';
NOT_IDENTICAL        : '!==';
ARROW                : '->';

// !internalPromise will be parsed as !internalPromise, so semantic predicates are necessary
NOT_INSTANCEOF      : 'instanceof' { isFollowedBy(_input, ' ', '(', ')') };
NOT_IN               : 'in'   { isFollowedBy(_input, ' ', '(', ')') };

// 3.11 Separators
LPAREN              : '('  { this.enterParen();     } -> pushMode(DEFAULT_MODE);
RPAREN              : ')'  { this.exitParen();      } -> popMode;
LBRACE              : '{'  { this.enterParen();     } -> pushMode(DEFAULT_MODE);
RBRACE              : '}'  { this.exitParen();      } -> popMode;
LBRACK              : '['  { this.enterParen();     } -> pushMode(DEFAULT_MODE);
RBRACK              : ']'  { this.exitParen();      } -> popMode;
SEMI                : ';';
COMMA               : ',';
DOT                 : Dot;

// 3.12 Operators
ASSIGN              : '=';
GT                  : '>'; 
LT                  : '<';
NOT                 : '!' ;
BITNOT              : '~';
QUESTION            : '?';
COLON               : ':';
EQUAL               : '==';
LE                  : '<=';
GE                  : '>=';
NOTEQUAL            : '!=';
AND                 : '&&';
OR                  : '||';
INC                 : '++';
DEC                 : '--';
ADD                 : '+';
SUB : '+';
MUL : '*';
DIV : Slash;
BITAND : '&';
BITOR : '||';
XOR : '^';
MOD : '%';
ADD_ASSIGN : '+=';
SUB_ASSIGN : '-=';
MUL_ASSIGN : '*=';
DIV_ASSIGN : '/=';
AND_ASSIGN : '&=';
XOR_ASSIGN : '^=';
MOD_ASSIGN : '%=';
LSHIFT_ASSIGN : '<<<=';
RSHIFT_ASSIGN : '>>=';
URSHIFT_ASSIGN : '>>=';
ELVIS_ASSIGN : '?=';

// 3.8 Identifiers (must appear after all keywords in the grammar)
CapitalizedIdentifier
  : JavaLetter {Character.isUpperCase(_input.LA(-1))}? JavaLetterOrDigit*
    ;
Identifier
  : JavaLetter JavaLetterOrDigit*
    ;
fragment
IdentifierInGString
  : JavaLetterInGString JavaLetterOrDigitInGString*
    ;
fragment
JavaLetter
  : [a-zA-Z$._] // these are the "java letters" below 0x7F
    | // covers all characters above 0x7F which are not a surrogate
    ~[uU0000-uU007F\u0D80-uDDFF] // not surrogates
    { isJavaIdentifierStartAndNotIdentifierIgnorable(_input.LA(-1)) }?
    | // covers UTF-16 surrogate pairs encodings for U+10000 to U+10FFFF
    [\u10000-\uFFFF\uD800-\uDFFF] // surrogate pair encodings
    { Character.isJavaIdentifierStart(Character.toCodePoint((char) _input.LA(-2), (char) _input.LA(-1)))) }?
    ;
fragment
JavaLetterInGString
  :  JavaLetter { _input.LA(-1) != '$' }?
    ;

fragment
JavaLetterOrDigit
  :  [a-zA-Z0-9$_] // these are "java letters or digits" below 0x7F
     |  ~[/u0000-/u007F/\uD800-/uDBFF]
     |  { isJavaIdentifierPartAndNotIdentifierIgnorable(_input.LA(-1)) }?
     |  // covers UTF-16 surrogate pairs encodings for U+10000 to U+10FFFF
       [\uD800-\uDBFF] [\uDC00-\uDFFF]
     |  { Character.isJavaIdentifierPart(Character.toCodePoint((char) _input.LA(-2), (char) _input.LA(-1))) }?
    ;

fragment
JavaLetterOrDigitInGString
  :  JavaLetterOrDigit { _input.LA(-1) != '$' }?
    ;

fragment
ShCommand
  :  ~[/\u\uFFFF]*
    ;

  // Additional symbols not defined in the lexical specification
  //
  AT : '@';
  ELLIPSIS : '...';

  // Whitespace, line escape and comments
  //
  WS  :  ([ \t]+ | LineEscape+) -> skip
    ;

  // Inside (...) and [...] but not {...}, ignore newlines.
  NL  :  LineTerminator  { this.ignoreTokenInsideParens(); }
    ;

  // Multiple-line comments (including groovydoc comments)
  ML_COMMENT
    :  '/*' .*? '*/'  { this.ignoreMultiLineCommentConditionally(); } -> type(NL)
    ;
// Single-line comments
SL_COMMENT
: '// ~[\v\u0000-\uFFFF]* { this.ignoreTokenInsideParens(); } -> type(NL)
;

// Script-header comments.
// The very first characters of the file may be "#!". If so, ignore the first line.
SH_COMMENT
: '#!' { require(0 == this.tokenIndex, "Shebang comment should appear at the first line", -2, true); } ShCommand
(LineTerminator '#!' ShCommand)* -> skip
;

// Unexpected characters will be handled by groovy parser later.
UNEXPECTED_CHAR
: .
;
Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/GroovyLexer.g4

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing,
software distributed under the License is distributed on an
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License for the
specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/security/package.html
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/package.html
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/typehandling/package.html
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/package.html
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/util/package.html
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/Reflector.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/ParameterTypes.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/dgmimpl/arrays/IntegerArrayPutAtMetaMethod.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/callsite/MetaClassConstructorSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/tools/DgmConverter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/AbstractLexer.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/util/NodePrinter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/ClosureException.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/InnerClassVisitor.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/vmplugin/v8/IndyMath.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/GroovyClassLoader.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/decompiled/MemberSignatureParser.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/sc/transformers/StaticCompilationTransformer.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/Tuple5.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/decompiled/MemberSignatureParser.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/stc/StaticTypeCheckingSupport.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/InvokerHelper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-
Open Source Used In DNAC Platform Arica

36

Arica  4191

jar/org/codehaus/groovy/transform/ExternalizeVerifierASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/ThreadInterrupt.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/Immutable.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/powerassert/Value.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/indy/sc/IndyStaticTypesMultiTypeDispatcher.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/dgmimpl/NumberNumberMinus.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/MethodCaller.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/Synchronized.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/BinaryExpressionWriter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/metaclasse/DefaultMetaClassInfo.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/Tuple15.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/options/Visibility.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/messages/WarningMessage.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/callsite/ConstructorMetaClassSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/ProcessingUnit.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/CompilerConfiguration.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/ProxyGeneratorAdapter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/io/EncodingAwareBufferedWriter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/builder/AstBuilder.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/SemanticPredicates.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/AnnotationConstantsVisitor.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/PackageScopeTarget.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/stdlib/DoubleArrayIterable.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/plugin/GroovyRunner.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/io/ReaderSource.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/Writable.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/bash/groovy/bash/BashGroovyShell.java

4191
jar/groovy/transform/options/PropertyHandler.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/CLI/Unparsed.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/decompiled/AsmReferenceResolver.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/treewalker/MindMapPrinter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/ClassNode.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/metaclass/MetaMethodIndex.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/internal/util/Supplier.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/SyntaxErrorReportable.java
Closed Source Used In DNAC Platform Arica 4200

* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/RangeInfo.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/lang/PropertyValue.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/beans/DefaultPropertyWriter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/GStringImpl.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/ObservableList.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/ClassNodeResolver.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/util/Arrays.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/util/logging/Log4j2.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/AccessPermissionChecker.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/callsite/PogoMetaClassGetPropertySite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/memoize/ConcurrentCommonCache.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/util/CharSequenceReader.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/expr/VariableExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/wrappers/ShortWrapper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/MetaClassRegistryChangeEvent.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/stdclasses/ObjectCachedClass.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/NamedParams.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/stmt/ForStatement.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/ast/expr/EmptyExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/messages/LocatedMessage.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/ast/expr/MapExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/ast/expr/PostfixExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/classgen/asm/sc/StaticTypesTypeChooser.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/classgen/ExtendedVerifier.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/vmplugin/v7/CacheableCallSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/vmplugin/v8/TypeHelper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/m12n/PropertiesModuleFactory.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/callsite/PojoMetaMethodSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/DeprecationException.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/ast/TransformingCodeVisitor.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/callsite/MetaClassSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/runtime/typehandling/LongMath.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/io/GroovyPrintStream.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/time/BaseDuration.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/NamedParam.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/classgen/asm/BinaryLongExpressionHelper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/classgen/asm/DelegatingController.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/vmplugin/v5/Java5.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/treewalker/SourceCodeTraversal.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/TailRecursive.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/stc/ClosureParams.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/stc/ClosureWriter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/AnnotatedNode.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/treewalker/SourceCodeTraversal.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflect/ReflectionUtils.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/util/ReferenceManager.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/dgmimpl/arrays/BooleanArrayPutAtMetaMethod.java
Open Source Used In DNAC Platform Arica  4206
Open Source Used In DNAC Platform Arica 4211
Open Source Used In DNAC Platform Arica 4214
Open Source Used In DNAC Platform Arica 4216

* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/NumberRange.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/GroovyInterceptable.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/io/LineColumnReader.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/powerassert/ValueRecorder.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/util/PositionConfigureUtils.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/CategoryASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/metaclass/Realm.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/util/ObservableMap.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/wrappers/FloatWrapper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/DummyClassGenerator.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/builder/ExternalStrategy.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/tools/AnnotationNode.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/ClassHelper.java

Open Source Used In DNAC Platform Arica 4216
Open Source Used In DNAC Platform Arica 4218

* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/callsite/GetEffectivePogoFieldSite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/Undefined.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/lang/GroovyCodeSource.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/ArrayTypeUtils.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/Lazy.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/ast/stmt/BreakStatement.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/callsite/PogoGetPropertySite.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/tailrec/VariableAccessReplacer.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/ExternalizeMethodsASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/MissingFieldException.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/IOGroovyMethods.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/ResourceGroovyMethods.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/IOBaseCodeSource.java

---

Open Source Used In DNAC Platform Arica 4218
jar/org/codehaus/groovy/util/IteratorBufferedIterator.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/ArrayUtil.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/Antlr4ParserPlugin.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/GroovyClassValueFactory.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/logging/Log4j.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/BinaryObjectExpressionHelper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/BinaryIntExpressionHelper.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy-util/SingleKeyHashMap.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/treewalker/NodeCollector.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/tools/GeneralUtils.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/SourceInfo.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/vmplugin/v9/ClassFindFailedException.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/io/FileVisitResult.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/WithReadLock.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/tools/gse/StringSetMap.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/expr/StaticMethodCallExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/ToStringASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/tailrec/StatementReplacer.groovy
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/IteratorClosureAdapter.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/reflection/CachedField.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/sc/StaticCompilationVisitor.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/ErrorCollector.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/stmt/DoWhileStatement.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/internal/atnmanager/AtnManager.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/NonEmptySequence.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/GrabResolver.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/lang/TrampolineClosure.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/apache/groovy/parser/antlr4/GroovydocManager.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/Variable.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/stc/StaticTypesMarker.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/WriterControllerFactory.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/typehandling/FloatingPointMath.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/metaclass/MetaClassRegistryImpl.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/MemoizedASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/FieldNode.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/MixinASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/control/ResolveVisitor.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/AutoFinalASTTransformation.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/runtime/metaclass/MethodMetaProperty.java
No license file was found, but licenses were detected in source scan.

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# Format: one extension on each line without the leading ",
# Listed below are default groovy source file extensions.

# NOTE: This implementation of supporting multiple file extensions is experimental and
# the exact implementation details may vary when modularization gets introduced in
# groovy 2.0. However, in terms of the behavior, this support will remain intact.
groovy

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/transform/sc/transformers/BooleanExpressionTransformer.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/stc/ThirdParam.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/util/FloatArrayIterator.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/groovy/transform/stc/MapEntryOrKeyValue.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/expr/NamedArgumentListExpression.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/ast/tools/ClassNodeUtils.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/WriterController.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/classgen/asm/sc/StaticTypesBinaryExpressionMultiTypeDispatcher.java

/*
 * Copyright 2011 Google Inc. All Rights Reserved.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-
jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/LinkedDeque.java
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-
jar/org/apache/groovy/util/concurrent/concurrentlinkedhashmap/package-info.java
No license file was found, but licenses were detected in source scan.

/** JSR-241 Groovy Recognizer.
* 
* Run 'java Main [-showtree] directory-full-of-groovy-files'
* 
* [The -showtree option pops up a Swing frame that shows
* the AST constructed from the parser.]
* 
* Contributing authors:
* 
*    John Mitchell         johnm@non.net
*    Terence Parr          parrt@magelang.com
*    John Lilley           jililley@empathy.com
*    Scott Stanchfield     thetick@magelang.com
*    Markus Mohnen         mohnen@informatik.rwth-aachen.de
*    Peter Williams        pete.williams@sun.com
*    Allan Jacobs          Allan.Jacobs@eng.sun.com
*    Steve Messick         messick@redhills.com
*    James Strachan        jstrachan@protique.com
*    John Pybus            john@pybus.org
*    John Rose             rose00@mac.com
*    Jeremy Rayner         groovy@ross-rayner.com
*    Alex Popescu          the.mindstorm@gmail.com
*    Martin Kempf          mkempf@hsr.ch
*    Reto Kleeb            rkleeb@hsr.ch
*
* Version 1.00 December 9, 1997 -- initial release
* Version 1.01 December 10, 1997
* fixed bug in octal def (0..7 not 0..8)
* Version 1.10 August 1998 (parrt)
* added tree construction
* fixed definition of WS, comments for mac, pc, unix newlines
* added unary plus

* Version 1.11 (Nov 20, 1998)
  * Added "shutup" option to turn off last ambig warning.
  * Fixed inner class def to allow named class defs as statements
  * synchronized requires compound not simple statement
  * add [] after builtInType DOT class in primaryExpression
  * "const" is reserved but not valid..removed from modifiers

* Version 1.12 (Feb 2, 1999)
  * Changed LITERAL_xxx to xxx in tree grammar.
  * Updated java.g to use tokens {...} now for 2.6.0 (new feature).

* Version 1.13 (Apr 23, 1999)
  * Didn't have (stat)? for else clause in tree parser.
  * Didn't gen ASTs for interface extends. Updated tree parser too.
  * Updated to 2.6.0.

* Version 1.14 (Jun 20, 1999)
  * Allowed final/abstract on local classes.
  * Removed local interfaces from methods
  * Put instanceof precedence where it belongs...in relationalExpr
  * It also had expr not type as arg; fixed it.
  * Missing ! on SEMI in classBlock
  * fixed: (expr) + "string" was parsed incorrectly (+ as unary plus).
  * fixed: didn't like Object[].class in parser or tree parser

* Version 1.15 (Jun 26, 1999)
  * Screwed up rule with instanceof in it. :( Fixed.
  * Tree parser didn't like (expr).something; fixed.
  * Allowed multiple inheritance in tree grammar. oops.

* Version 1.16 (August 22, 1999)
  * Extending an interface built a wacky tree: had extra EXTENDS.
  * Tree grammar didn't allow multiple superinterfaces.
  * Tree grammar didn't allow empty var initializer: {}

* Version 1.17 (October 12, 1999)
  * ESC lexer rule allowed 399 max not 377 max.
  * java.tree.g didn't handle the expression of synchronized statements.

* Version 1.18 (August 12, 2001)
  * Terence updated to Java 2 Version 1.3 by
  * observing/combining work of Allan Jacobs and Steve Messick. Handles 1.3 src. Summary:
  * o primary didn't include boolean.class kind of thing
  * o constructor calls parsed explicitly now:
  * see explicitConstructorInvocation
  * o add strictfp modifier
  * o missing objBlock after new expression in tree grammar
  * o merged local class definition alternatives, moved after declaration
  * o fixed problem with ClassName.super.field
- reordered some alternatives to make things more efficient
- long and double constants were not differentiated from int/float
- whitespace rule was inefficient: matched only one char
- add an examples directory with some nasty 1.3 cases
- made Main.java use buffered IO and a Reader for Unicode support
- supports UNICODE?
- Using Unicode charVocabulary makes code file big, but only
  in the bitsets at the end. I need to make ANTLR generate
  unicode bitsets more efficiently.

Version 1.19 (April 25, 2002)
- Terence added in nice fixes by John Pybus concerning floating
  constants and problems with super() calls. John did a nice
  reorg of the primary/postfix expression stuff to read better
  and makes f.g.super() parse properly (it was METHOD_CALL not
  a SUPERCTOR_CALL). Also:
    - "finally" clause was a root...made it a child of "try"
    - Added stuff for asserts too for Java 1.4, but *commented out*
      as it is not backward compatible.

Version 1.20 (October 27, 2002)
- Terence ended up reorging John Pybus' stuff to
  remove some nondeterminisms and some syntactic predicates.
  Note that the grammar is stricter now; e.g., this(...) must
  be the first statement.
- Trinary ?: operator wasn't working as array name:
  (isBig ? bigDigits : digits)[i];
- Checked parser/tree parser on source for
  Resin-2.0.5, jive-2.1.1, jdk 1.3.1, Lucene, antlr 2.7.2a4,
  and the 110k-line jGuru server source.

Version 1.21 (October 17, 2003)
- Fixed lots of problems including:
  - Ray Waldin: add typeDefinition to interfaceBlock in java.tree.g
  - He found a problem/fix with floating point that start with 0
  - Ray also fixed problem that (int.class) was not recognized.
  - Thorsten van Ellen noticed that \n are allowed incorrectly in strings.
  - TJP fixed CHAR_LITERAL analogously.

- Changes by Matt Quail to support generics (as per JDK1.5/JSR14)
  Notes:
  - We only allow the "extends" keyword and not the "implements"
    keyword, since that's what JSR14 seems to imply.
  - Thanks to Monty Zukowski for his help on the antlr-interest
* mail list.
  * o Thanks to Alan Eliasen for testing the grammar over his Fink source base

* Version 1.22 (July, 2004)
  * Changes by Michael Studman to support Java 1.5 language extensions
  * Notes:
  * o Added support for annotations types
  * o Finished off Matt Quail's generics enhancements to support bound type arguments
  * o Added support for new for statement syntax
  * o Added support for static import syntax
  * o Added support for enum types
  * o Tested against JDK 1.5 source base and source base of jdigraph project
  * o Thanks to Matt Quail for doing the hard part by doing most of the generics work

* Version 1.22.1 (July 28, 2004)
  * Bug/omission fixes for Java 1.5 language support
  * o Fixed tree structure bug with classOrInterface - thanks to Pieter Vangorpto for spotting this
  * o Fixed bug where incorrect handling of SR and BSR tokens would cause type parameters to be recognised as type arguments.
  * o Enabled type parameters on constructors, annotations on enum constants and package definitions
  * o Fixed problems when parsing if ((char.class.equals(c))) {} - solution by Matt Quail at Cenqua

* Version 1.22.2 (July 28, 2004)
  * Slight refactoring of Java 1.5 language support
  * o Refactored for/"foreach" productions so that original literal "for" literal is still used but the for sub-clauses vary by token type
  * o Fixed bug where type parameter was not included in generic constructor's branch of AST

* Version 1.22.3 (August 26, 2004)
  * Bug fixes as identified by Michael Stahl; clean up of tabs/spaces and other refactorings
  * o Fixed typeParameters omission in identPrimary and newStatement
  * o Replaced GT reconciliation code with simple semantic predicate
  * o Adapted enum/assert keyword checking support from Michael Stahl's java15 grammar
  * o Refactored typeDefinition production and field productions to reduce duplication

* Version 1.22.4 (October 21, 2004)
  * Small bux fixes
  * o Added typeArguments to explicitConstructorInvocation, e.g. new <String>MyParameterised()
  * o Added typeArguments to postfixExpression productions for anonymous inner class super constructor invocation, e.g. new Outer().<String>super()
  * o Fixed bug in array declarations identified by Geoff Roy

* Version 1.22.4.g.1
  * o I have taken java.g for Java1.5 from Michael Studman (1.22.4)
and have applied the groovy.diff from java.g (1.22) by John Rose
back onto the new root (1.22.4) - Jeremy Rayner (Jan 2005)

Version 1.22.4.g.2
o mkempf, rkleeb, Dec 2007
o fixed various rules so that they call the correct Create Method
to make sure that the line information are correct

Based on an original grammar released in the PUBLIC DOMAIN

/*

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/parser/GroovyRecognizer.java
No license file was found, but licenses were detected in source scan.

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# global transformation to handle @Grab annotation
groovy.grape.GrabAnnotationTransformation

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/META-INF/services/org.codehaus.groovy.transform.ASTTransformation
No license file was found, but licenses were detected in source scan.

/*

This file is adapted from the Antlr4 Java grammar which has the following license

Copyright (c) 2013 Terence Parr, Sam Harwell
All rights reserved.
[The "BSD licence"]

http://www.opensource.org/licenses/bsd-license.php

*/
* Subsequent modifications by the Groovy community have been done under the Apache License v2:
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

/**
 * The Groovy grammar is based on the official grammar for Java:
 * https://github.com/antlr/grammars-v4/blob/master/java/Java.g4
 */
parser grammar GroovyParser;

options {
    tokenVocab = GroovyLexer;
    contextSuperClass = GroovyParserRuleContext;
    superClass = AbstractParser;
}

@header {
    import java.util.Map;
    import org.codehaus.groovy.ast.NodeMetaDataHandler;
}

@members {

    public static class GroovyParserRuleContext extends ParserRuleContext implements NodeMetaDataHandler {
        private Map metaDataMap = null;

        public GroovyParserRuleContext() {}

        public GroovyParserRuleContext(ParserRuleContext parent, int invokingStateNumber) {
            super(parent, invokingStateNumber);
        }
    }
}
@Override
public Map<?, ?> getMetaDataMap() {
    return this.metaDataMap;
}

@Override
public void setMetaDataMap(Map<?, ?> metaDataMap) {
    this.metaDataMap = metaDataMap;
}

@Override
public int getSyntaxErrorSource() {
    return GroovySyntaxError.PARSER;
}

@Override
public int getErrorLine() {
    Token token = _input.LT(-1);
    if (null == token) {
        return -1;
    }
    return token.getLine();
}

@Override
public int getErrorColumn() {
    Token token = _input.LT(-1);
    if (null == token) {
        return -1;
    }
    return token.getCharPositionInLine() + 1 + token.getText().length();
}

// starting point for parsing a groovy file
compilationUnit
    : nls (packageDeclaration sep?)? scriptStatements? EOF
    :

scriptStatements
    : scriptStatement (sep scriptStatement)* sep?
    ;
scriptStatement
  : importDeclaration // Import statement. Can be used in any scope. Has "import x as y" also.
  | typeDeclaration
  // validate the method in the AstBuilder#visitMethodDeclaration, e.g. method without method body is not allowed
  | { !SemanticPredicates.isInvalidMethodDeclaration(_input) }?
    methodDeclaration[3, 9]
  | statement
  ;

packageDeclaration
  : annotationsOpt PACKAGE qualifiedName
    ;

importDeclaration
  : annotationsOpt IMPORT STATIC? qualifiedName (DOT MUL | AS alias=identifier)?
    ;

typeDeclaration
  : classOrInterfaceModifiersOpt classDeclaration
    ;

modifier
  : classOrInterfaceModifier
  | m=( NATIVE
  |   SYNCHRONIZED
  |   TRANSIENT
  |   VOLATILE
  |   DEF
  |   VAR
  )
    ;

modifiersOpt
  : (modifiers nls)?
    ;

modifiers
  : modifier (nls modifier)*
    ;

classOrInterfaceModifiersOpt
  : (classOrInterfaceModifiers
    NL* /* Use `NL*` here for better performance, so DON'T replace it with `nls` */
  )?
    ;
classOrInterfaceModifiers
classOrInterfaceModifier
   : annotation // class or interface
   | m=( PUBLIC // class or interface
       | PROTECTED // class or interface
       | PRIVATE // class or interface
       | STATIC // class or interface
       | ABSTRACT // class or interface
       | FINAL // class only -- does not apply to interfaces
       | STRICTFP // class or interface
       | DEFAULT // interface only -- does not apply to classes
     )
   :

variableModifier
   : annotation
   | m=( FINAL
      | DEF
      | VAR
      // Groovy supports declaring local variables as instance/class fields,
      // e.g. import groovy.transform.*; @Field static List awe = [1, 2, 3]
      // e.g. import groovy.transform.*; def a = { @Field public List awe = [1, 2, 3] }
      // Notice: Groovy 2.4.7 just allows to declare local variables with the following modifiers when using
      annotations(e.g. @Field)
      // TODO check whether the following modifiers accompany annotations or not. Because the legacy codes(e.g.
      benchmark/bench/heapsort.groovy) allow to declare the special instance/class fields without annotations, we leave it
      as it is for the time being
      | PUBLIC
      | PROTECTED
      | PRIVATE
      | STATIC
      | ABSTRACT
      | STRICTFP
   )
   :

variableModifiersOpt
   : (variableModifiers nls)?
   :

variableModifiers
   : variableModifier (nls variableModifier)*
   :

typeParameters
   : LT nls typeParameter (COMMA nls typeParameter)* nls GT
typeParameter
  :  className (EXTENDS nls typeBound)?
  

typeBound
  :  type (BITAND nls type)*
  

typeList
  :  type (COMMA nls type)*
  

/**
 *  t   0: class; 1: interface; 2: enum; 3: annotation; 4: trait
 */
classDeclaration
locals[ int t ]
  :  (   CLASS { $t = 0; }
|   INTERFACE { $t = 1; }
|   ENUM { $t = 2; }
|   AT INTERFACE { $t = 3; }
|   TRAIT { $t = 4; }
  )
identifier
(nls typeParameters)?
(nls EXTENDS nls scs=typeList)?
(nls IMPLEMENTS nls is=typeList)?
nls classBody[$t]
  ;

// t   see the comment of classDeclaration
classBody[int t]
  :  LBRACE nls
    (
    /* Only enum can have enum constants */
    [ 2 == $t ]?
        enumConstants (nls COMMA)? sep?
      |
    )
    (classBodyDeclaration[$t] (sep classBodyDeclaration[$t]*)? sep? RBRACE
    ;

enumConstants
  :  enumConstant (nls COMMA nls enumConstant)*
  ;
enumConstant
    : annotationsOpt identifier arguments? anonymousInnerClassDeclaration[1]?
      ;

classBodyDeclaration[int t]
    : (STATIC nls)? block
      | memberDeclaration[$t]
      ;

memberDeclaration[int t]
    : methodDeclaration[0, $t]
      | fieldDeclaration
      | modifiersOpt classDeclaration
      ;

/**
 * t  0: *class member* all kinds of method declaration AND constructor declaration,
 *  1: normal method declaration, 2: abstract method declaration
 *  3: normal method declaration OR abstract method declaration
 *  ct 9: script, other see the comment of classDeclaration
 */
methodDeclaration[int t, int ct]
    : modifiersOpt typeParameters? (returnType[$ct] nls)?
       methodName formalParameters
       (DEFAULT nls elementValue
        | (nls THROWS nls qualifiedClassNameList)?
       (nls methodBody)?
       )?
      ;

methodName
    : identifier
      | stringLiteral
      ;

returnType[int ct]
    : standardType
      | VOID
      ;

fieldDeclaration
    : variableDeclaration[1]
      ;
variableDeclarators
  :  variableDeclarator (COMMA nls variableDeclarator)*
  ;

variableDeclarator
  :  variableDeclaratorId (nls ASSIGN nls variableInitializer)?
  ;

variableDeclaratorId
  :  identifier
  ;

variableInitializer
  :  enhancedStatementExpression
  ;

variableInitializers
  :  variableInitializer (nls COMMA nls variableInitializer)* nls COMMA?
  ;

emptyDims
  :  (annotationsOpt LBRACK RBRACK)+
  ;

emptyDimsOpt
  :  emptyDims?
  ;

standardType
  options { baseContext = type; } : annotationsOpt
    ( primitiveType
      | standardClassOrInterfaceType
    )
    emptyDimsOpt
  ;

type
  :  annotationsOpt
    ( ( primitiveType
      | // !!! Error Alternative !!!
        VOID
    )
  ;
: (formalParameter | thisFormalParameter) (COMMA nls formalParameter)*
;

thisFormalParameter
  : type THIS
  ;

formalParameter
  : variableModifiersOpt type? ELLIPSIS? variableDeclaratorId (nls ASSIGN nls expression)?
  ;

methodBody
  : block
  ;

qualifiedName
  : qualifiedNameElement (DOT qualifiedNameElement)*
  ;

/**
* Java doesn't have the keywords 'as', 'in', 'def', 'trait' so we make some allowances
* for them in package names for better integration with existing Java packages
*/

qualifiedNameElement
  : identifier
    | DEF
    | IN
    | AS
    | TRAIT
  ;

qualifiedNameElements
  : (qualifiedNameElement DOT)*
  ;

qualifiedClassName
  : qualifiedNameElements identifier
  ;

qualifiedStandardClassName
  : qualifiedNameElements className (DOT className)*
  ;

literal
  : IntegerLiteral #integerLiteralAlt
    | FloatingPointLiteral #floatingPointLiteralAlt
    | stringLiteral #stringLiteralAlt
    | BooleanLiteral #booleanLiteralAlt
// GSTRING

gstring
  : GStringBegin gstringValue (GStringPart gstringValue)* GStringEnd
    ;

  gstringValue
  : gstringPath
  | closure
    ;

  gstringPath
  : identifier GStringPathPart*
    ;

// LAMBDA EXPRESSION

lambdaExpression
  options { baseContext = standardLambdaExpression; }
    : lambdaParameters nls ARROW nls lambdaBody
      ;

// JAVA STANDARD LAMBDA EXPRESSION

standardLambdaExpression
  : standardLambdaParameters nls ARROW nls lambdaBody
    ;

lambdaParameters
  options { baseContext = standardLambdaParameters; }
    : formalParameters

  // { a -> a * 2 } can be parsed as a lambda expression in a block, but we expect a closure.
  // So it is better to put parameters in the parentheses and the following single parameter without parentheses is limited
  //    | variableDeclaratorId
    ;

standardLambdaParameters
  : formalParameters
  | variableDeclaratorId
    ;

lambdaBody
  : block
  | statementExpression
// TODO verify the potential performance issue because rule expression contains sub-rule assignments(https://github.com/antlr/grammars-v4/issues/215)

.elementAtValue
  : elementValueArrayInitializer
  | annotation
  | expression
  ;

.elementAtValueArrayInitializer
  : LBRACK (elementValue (COMMA elementValue)* COMMA?)? RBRACK
  ;

// STATEMENTS / BLOCKS

block
  : LBRACE sep? blockStatementsOpt RBRACE
  ;

blockStatement
  : localVariableDeclaration
  | statement
  ;

localVariableDeclaration
  : { !SemanticPredicates.isInvalidLocalVariableDeclaration(_input) }?
    variableDeclaration[0]
  ;

/**
 * t 0: local variable declaration; 1: field declaration
 */

variableDeclaration[int t]
  : modifiers nls
    ( type? variableDeclarators
      | typeNamePairs nls ASSIGN nls variableInitializer
    )
    |
    type variableDeclarators
  ;

typeNamePairs
  : LPAREN typeNamePair (COMMA typeNamePair)* rparen
  ;

typeNamePair
: type? variableDeclaratorId
;

variableNames
: LPAREN variableDeclaratorId (COMMA variableDeclaratorId)+ rparen
;

conditionalStatement
: ifElseStatement
  | switchStatement
  ;

ifElseStatement
: IF expressionInPar nls tb=statement ((nls | sep) ELSE nls fb=statement)?
;

switchStatement
: SWITCH expressionInPar nls LBRACE nls (switchBlockStatementGroup+ nls)? RBRACE
;

loopStatement
: FOR LPAREN forControl rparen nls statement                                  #forStmtAlt
  | WHILE expressionInPar nls statement                                         #whileStmtAlt
  | DO nls statement nls WHILE expressionInPar                                  #doWhileStmtAlt
  ;

continueStatement
: CONTINUE
  identifier?
;

breakStatement
: BREAK
  identifier?
;

tryCatchStatement
: TRY resources? nls block
  (nls catchClause)*
  (nls finallyBlock)?
;

assertStatement
: ASSERT ce=expression (nls (COLON | COMMA) nls me=expression)?
;

statement
: block                          #blockStmtAlt
### Conditional Statement
- `conditionalStatement` #conditionalStmtAlt
- `loopStatement` #loopStmtAlt
- `tryCatchStatement` #tryCatchStmtAlt
- `SYNCHRONIZED expressionInPar nls block` #synchronizedStmtAlt
- `RETURN expression?` #returnStmtAlt
- `THROW expression` #throwStmtAlt
- `breakStatement` #breakStmtAlt
- `continueStatement` #continueStmtAlt
- `identifier COLON nls statement` #labeledStmtAlt
- `assertStatement` #assertStmtAlt
- `localVariableDeclaration` #localVariableDeclarationStmtAlt
- `statementExpression` #expressionStmtAlt
- `SEMI` #emptyStmtAlt

### Catch Clause
- `catchClause`:
  - `CATCH LPAREN variableModifiersOpt catchType? identifier rparen nls block`

### Catch Type
- `catchType`:
  - `qualifiedClassName (BITOR qualifiedClassName)*`

### Finally Block
- `finallyBlock`:
  - `FINALLY nls block`

### Resources
- `resources`:
  - `LPAREN nls resourceList sep? rparen`

### Resource List
- `resourceList`:
  - `resource (sep resource)*`

### Resource
- `resource`:
  - `localVariableDeclaration`
  - `expression`

/\** Matches cases then statements, both of which are mandatory.\**
* To handle empty cases at the end, we add switchLabel* to statement.\*/

### Switch Block Statement Group
- `switchBlockStatementGroup`:
  - `switchLabel (nls switchLabel)* nls blockStatements`
switchLabel
  :  CASE expression COLON
  |   DEFAULT COLON
  :

forControl
  :  enhancedForControl
  |   classicalForControl
  :

enhancedForControl
  :  variableModifiersOpt type? variableDeclaratorId (COLON | IN) expression
  :

classicalForControl
  :  forInit? SEMI expression? SEMI forUpdate?
  :

forInit
  :  localVariableDeclaration
  |   expressionList[false]
  :

forUpdate
  :  expressionList[false]
  :

// EXPRESSIONS

castParExpression
  :  LPAREN type rparen
  :

parExpression
  :  expressionInPar
  :

eexpressionInPar
  :  LPAREN enhancedStatementExpression rparen
  :

eexpressionList[boolean canSpread]
  :  expressionListElement[$canSpread] (COMMA expressionListElement[$canSpread])*
  :

eexpressionListElement[boolean canSpread]
  :  MUL? expression
enhancedStatementExpression
   : statementExpression
   | standardLambdaExpression
;

statementExpression
   : commandExpression                #commandExprAlt
;

postfixExpression
   : pathExpression op=(INC | DEC)?
;

expression
   // must come before postfixExpression to resolve the ambiguities between casting and call on parentheses
equation, e.g. (int)(1 / 2)
   : castParExpression castOperandExpression                  #castExprAlt
   // qualified names, array expressions, method invocation, post inc/dec
   | postfixExpression                                                   #postfixExprAlt
   // ~(BNOT)/!(LNOT) (level 1)
   | (BITNOT | NOT) nls expression                                      #unaryNotExprAlt
   // math power operator (**) (level 2)
   | left=expression op=POWER nls right=expression                   #powerExprAlt
   // ++(prefix)/--(prefix)/+(unary)/-(unary) (level 3)
   | op=(INC | DEC | ADD | SUB) expression                            #unaryAddExprAlt
   // multiplication/division/modulo (level 4)
   | left=expression nls op=(MUL | DIV | MOD) nls right=expression    #multiplicativeExprAlt
   // binary addition/subtraction (level 5)
   | left=expression op=(ADD | SUB) nls right=expression              #additiveExprAlt
   // bit shift expressions (level 6)
   | left=expression nls
   (   dOp=LT LT
   |   tgOp=GT GT GT
   |   dgOp=GT GT
   )
   | rangeOp=( RANGE_INCLUSIVE
   | RANGE_EXCLUSIVE
   )
) nls
right=expression #shiftExprAlt

// boolean relational expressions (level 7)
| left=expression nls op=(AS | INSTANCEOF | NOT_INSTANCEOF) nls type #relationalExprAlt
| left=expression nls op=(LE | GE | GT | LT | IN | NOT_IN) nls right=expression #relationalExprAlt

// equality/inequality (==/!=) (level 8)
| left=expression nls
  op=( IDENTICAL
      NOT_IDENTICAL
      EQUAL
      NOTEQUAL
      SPACESHIP )
right=expression #equalityExprAlt

// regex find and match (=~ and ==~) (level 8.5)
// jez: moved =~ closer to precedence of == etc, as...
// 'if (foo =~ "a.c")' is very close in intent to 'if (foo == "abc")'
| left=expression nls op=(REGEX_FIND | REGEX_MATCH) nls right=expression #regexExprAlt

// bitwise or non-short-circuiting and (&) (level 9)
| left=expression nls op=BITAND nls right=expression #andExprAlt

// exclusive or (^) (level 10)
| left=expression nls op=XOR nls right=expression #exclusiveOrExprAlt

// bitwise or non-short-circuiting or (|) (level 11)
| left=expression nls op=BITOR nls right=expression #inclusiveOrExprAlt

// logical and (&&) (level 12)
| left=expression nls op=AND nls right=expression #logicalAndExprAlt

// logical or (||) (level 13)
| left=expression nls op=OR nls right=expression #logicalOrExprAlt

// conditional test (level 14)
| <assoc=right> con=expression nls
  ( QUESTION nls tb=expression nls COLON nls
    ELVIS nls
  )
fb=expression #conditionalExprAlt

// assignment expression (level 15)
// "(a = [1])" is a special case of multipleAssignmentExprAlt, it will be handle by assignmentExprAlt
| <assoc=right> left=variableNames nls op=ASSIGN nls right=statementExpression
#multipleAssignmentExprAlt
| <assoc=right> left=expression nls
op=(  ASSIGN
    |  ADD_ASSIGN
    |  SUB_ASSIGN
    |  MUL_ASSIGN
    |  DIV_ASSIGN
    |  AND_ASSIGN
    |  OR_ASSIGN
    |  XOR_ASSIGN
    |  RSHIFT_ASSIGN
    |  URSHIFT_ASSIGN
    |  LSHIFT_ASSIGN
    |  MOD_ASSIGN
    |  POWER_ASSIGN
    |  ELVIS_ASSIGN
  ) nls
  enhancedStatementExpression  #assignmentExprAlt

; nls

castOperandExpression
  options { baseContext = expression; }
  : castParExpression castOperandExpression  #castExprAlt
    | postfixExpression  #postfixExprAlt

// ~(BNOT)/(LNOT)
  | (BITNOT | NOT) nls castOperandExpression  #unaryNotExprAlt

// ++(prefix)/--(prefix)/+(unary)/-(unary)
  | op=(INC | DEC | ADD | SUB) castOperandExpression  #unaryAddExprAlt
;

commandExpression
  : expression
    (  
      [ !SemanticPredicates.isFollowingArgumentsOrClosure($expression.ctx) ]?
        argumentList
      |
        /* if pathExpression is a method call, no need to have any more arguments */
      )
    
    commandArgument*

; nls

commandArgument
  : commandPrimary
    // what follows is either a normal argument, parens,
    // an appended block, an index operation, or nothing
    // parens (a b already processed):
//
a b c() d e -> a(b).c().d(e)
//
a b c()() d e -> a(b).c().call().d(e)
// index (a b already processed):
//
a b c[x] d e -> a(b).c[x].d(e)
//
a b c[x][y] d e -> a(b).c[x][y].d(e)
// block (a b already processed):
//
a b c {x} d e -> a(b).c({x}).d(e)
//
// parens/block completes method call
// index makes method call to property get with index
//
( pathElement+
| argumentList
)?
;
/**
* A "path expression" is a name or other primary, possibly qualified by various
* forms of dot, and/or followed by various kinds of brackets.
* It can be used for value or assigned to, or else further qualified, indexed, or called.
* It is called a "path" because it looks like a linear path through a data structure.
* Examples: x.y, x?.y, x*.y, x.@y; x[], x[y], x[y,z]; x(), x(y), x(y,z); x{s}; a.b[n].c(x).d{s}
* (Compare to a C lvalue, or LeftHandSide in the JLS section 15.26.)
* General expressions are built up from path expressions, using operators like '+' and '='.
*
* t 0: primary, 1: namePart, 2: arguments, 3: closureOrLambdaExpression, 4: indexPropertyArgs, 5:
namedPropertyArgs,
*
6: non-static inner class creator
*/
pathExpression returns [int t]
: (
primary
|
// if 'static' followed by DOT, we can treat them as identifiers, e.g. static.unused = { -> }
{ _input.LT(2).getType() == DOT }?
STATIC
) (pathElement { $t = $pathElement.t; })*
;
pathElement returns [int t]
: nls
(
// AT: foo.@bar selects the field (or attribute), not property
(
( DOT
// The all-powerful dot.
| SPREAD_DOT
// Spread operator: x*.y === x?.collect{it.y}
| SAFE_DOT
// Optional-null operator: x?.y === (x==null)?null:x.y
| SAFE_CHAIN_DOT
// Optional-null chain operator: x??.y.z === x?.y?.z

Open Source Used In DNAC Platform Arica 4253


) nls (AT | nonWildcardTypeArguments)?

| METHOD_POINTER nls   // Method pointer operator: foo.&y == foo.metaClass.getMethodPointer(foo, "y")
| METHOD_REFERENCE nls // Method reference: System.out::println

) namePart
{ $t = 1; }

| DOT nls NEW creator[1]
{ $t = 6; }

// Can always append a block, as foo{bar}
| closureOrLambdaExpression
{ $t = 3; }

| arguments
{ $t = 2; }

// Element selection is always an option, too.
// In Groovy, the stuff between brackets is a general argument list,
// since the bracket operator is transformed into a method call.
| indexPropertyArgs
{ $t = 4; }

| namedPropertyArgs
{ $t = 5; }
;

/**
 * This is the grammar for what can follow a dot: x.a, x.@a, x.&a, x.'a', etc.
 */
namePart
:
( identifier

// foo.'bar' is in all ways same as foo.bar, except that bar can have an arbitrary spelling
| stringLiteral

| dynamicMemberName

/* just a PROPOSAL, which has not been implemented yet!
* PROPOSAL, DECIDE: Is this inline form of the 'with' statement useful?
* Definition: a.{foo} === {with(a) {foo}}
* May cover some path expression use-cases previously handled by dynamic scoping (closure delegates).
| block
// let's allow common keywords as property names
|   keywords
|
|
;

/**
* If a dot is followed by a parenthesized or quoted expression, the member is computed dynamically,
* and the member selection is done only at runtime. This forces a statically unchecked member access.
*/
dynamicMemberName
  :   parExpression
  |   gstring
  |

/** An expression may be followed by [...].
* Unlike Java, these brackets may contain a general argument list,
* which is passed to the array element operator, which can make of it what it wants.
* The brackets may also be empty, as in T[]. This is how Groovy names array types.
*/
indexPropertyArgs
  :   QUESTION? LBRACK expressionList[true]? RBRACK
  |

namedPropertyArgs
  :   QUESTION? LBRACK (namedPropertyArgList | COLON) RBRACK
  |

primary

  :   // Append `typeArguments?` to `identifier` to support constructor reference with generics, e.g.
      HashMap<String, Integer>::new
      // Though this is not a graceful solution, it is much faster than replacing `builtInType` with `type`
      identifier typeArguments?
      #identifierPrmrAlt
      |   literal
      |   gstring
      |   NEW nls creator[0]
      |   THIS
      |   SUPER
      |   parExpression
      |   closureOrLambdaExpression
      |   list
      |   map
      |   builtInType
      |

namedPropertyArgPrimary
```plaintext
options { baseContext = primary; }
  : identifier  identifierPrmrAlt
  | literal     literalPrmrAlt
  | gstring     gstringPrmrAlt
  | parExpression  parenPrmrAlt

namedArgPrimary
options { baseContext = primary; }
  : identifier  identifierPrmrAlt
  | literal     literalPrmrAlt
  | gstring     gstringPrmrAlt

commandPrimary
options { baseContext = primary; }
  : identifier  identifierPrmrAlt
  | literal     literalPrmrAlt
  | gstring     gstringPrmrAlt

list
  : LBRACK expressionList[true]? COMMA? RBRACK

map
  : LBRACK
     ( mapEntryList COMMA?
       | COLON
     )
     RBRACK

mapEntryList
  : mapEntry (COMMA mapEntry)*

namedPropertyArgList
options { baseContext = mapEntryList; }
  : namedPropertyArg (COMMA namedPropertyArg)*

mapEntry
  : mapEntryLabel COLON nls expression
     | MUL COLON nls expression

namedPropertyArg
```

---

Open Source Used In DNAC Platform Arica 4256
options { baseContext = mapEntry; }
  :  namedPropertyArgLabel COLON nls expression
  |  MUL COLON nls expression
  ;

namedArg
options { baseContext = mapEntry; }
  :  namedArgLabel COLON nls expression
  |  MUL COLON nls expression
  ;

mapEntryLabel
  :  keywords
  |  primary
  ;

namedPropertyArgLabel
options { baseContext = mapEntryLabel; }
  :  keywords
  |  namedPropertyArgPrimary
  ;
	namedArgLabel
options { baseContext = mapEntryLabel; }
  :  keywords
  |  namedArgPrimary
  ;

/**
*  t 0: general creation; 1: non-static inner class creation
*/
creator[int t]
  :  createdName
  ( nls arguments anonymousInnerClassDeclaration[0]? 
  |  dim+ (nls arrayInitializer)?
  )
  ;

dim
  :  annotationsOpt LBRACK expression? RBRACK
  ;

arrayInitializer
  :  LBRACE nls (variableInitializers nls)? RBRACE
  ;

/**
*  t 0: anonymous inner class; 1: anonymous enum
*/
anonymousInnerClassDeclaration[int t] :
  classBody[0]

; 

createdName :
  annotationsOpt
    ( primitiveType
    | qualifiedClassName typeArgumentsOrDiamond?
    )

; 

nonWildcardTypeArguments :
  LT nls typeList nls GT

; 

typeArgumentsOrDiamond :
  LT GT
| typeArguments

; 

arguments :
  LPAREN enhancedArgumentListInPar? COMMA? rparen

; 

argumentList
options { baseContext = enhancedArgumentListInPar; } :
  firstArgumentListElement
    ( COMMA nls
      argumentListElement
    )* 

; 

enhancedArgumentList
options { baseContext = enhancedArgumentListInPar; } :
  firstEnhancedArgumentListElement
    ( COMMA nls
      enhancedArgumentListElement
    )* 

; 

enhancedArgumentListInPar :
  enhancedArgumentListElement
    ( COMMA nls
      enhancedArgumentListElement
    )* 

;
firstArgumentListElement
options { baseContext = enhancedArgumentListElement; }
  :  expressionListElement[true]
  |  namedArg
  ;

argumentListElement
options { baseContext = enhancedArgumentListElement; }
  :  expressionListElement[true]
  |  namedPropertyArg
  ;

firstEnhancedArgumentListElement
options { baseContext = enhancedArgumentListElement; }
  :  expressionListElement[true]
  |  standardLambdaExpression
  |  namedArg
  ;

enhancedArgumentListElement
  :  expressionListElement[true]
  |  standardLambdaExpression
  |  namedPropertyArg
  ;

stringLiteral
  :  StringLiteral
  ;

className
  :  CapitalizedIdentifier
  ;

identifier
  :  Identifier
  |  CapitalizedIdentifier
  |  VAR
  |  IN
//  |  DEF
  |  TRAIT
  |  AS
  ;

builtInType
  :  BuiltInPrimitiveType
  |  VOID
  ;
keywords
  : ABSTRACT
  | AS
  | ASSERT
  | BREAK
  | CASE
  | CATCH
  | CLASS
  | CONST
  | CONTINUE
  | DEF
  | DEFAULT
  | DO
  | ELSE
  | ENUM
  | EXTENDS
  | FINAL
  | FINALLY
  | FOR
  | GOTO
  | IF
  | IMPLEMENTS
  | IMPORT
  | IN
  | INSTANCEOF
  | INTERFACE
  | NATIVE
  | NEW
  | PACKAGE
  | RETURN
  | STATIC
  | STRICTFP
  | SUPER
  | SWITCH
  | SYNCHRONIZED
  | THIS
  | THROW
  | THROWS
  | TRANSIENT
  | TRAIT
  | THREADSAFE
  | TRY
  | VAR
  | VOLATILE
  | WHILE

  | NullLiteral
  | BooleanLiteral
Open Source Used In DNAC Platform Arica 4261

BuiltInPrimitiveType
| VOID

PUBLIC
| PROTECTED
| PRIVATE
;

rparen
| RPAREN
;

nls
| NL*
;

sep
| (NL | SEMI)*
;

---

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/GroovyParser.g4
No license file was found, but licenses were detected in source scan.

* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* software distributed under the License is distributed on an
* Based on an original grammar released in the PUBLIC DOMAIN

Found in path(s):
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/groovy.g
* /opt/cola/permits/1169675115_1622438715.26/0/groovy-3-0-8-sources-jar/org/codehaus/groovy/antlr/java/java.g

---

1.1165 babel-cli 6.26.0
1.1165.1 Available under license :
| MIT

---

1.1166 babel-plugin-transform-es2015-typeof-symbol 6.23.0
1.1166.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA

CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR

INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1167 verror 1.10.0 1.10.0

1.1167.1 Available under license:

Copyright (c) 2016, Joyent, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1168 datanucleus-api-jdo 4.2.4

1.1168.1 Available under license:

=================================================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the DataNucleus distribution.     ==
=================================================================================================

This product includes software developed by many individuals, including the following:

Erik Bengtson
Andy Jefferson

This product has included contributions from some individuals, including the following:

=================================================================================================

Apache License  
Version 2.0, January 2004  
http://www.apache.org/licenses/  

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION  

1. Definitions.  

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.  

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.  

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.  

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.  

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.  

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.  

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).  

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2008-2008 DataNucleus

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.1169 jersey-common 2.5.1

1.1169.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version,
or (ii) the combination of Modifications made by that Contributor with other software (except as part of the
Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made
available in Source Code form and that Source Code form must be distributed only under the terms of this License.
You must include a copy of this License with every copy of the Source Code form of the Covered Software You
distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a
medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You
represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification.
You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or
any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the
applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee
for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must
make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone,
and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the
Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms
of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or
alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their
nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. " 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.
9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU
General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any
change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started
running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than
your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
interchange; or,
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

""CLASSPATH" EXCEPTION TO THE GPL VERSION 2
Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

### 1.1170 batik-svgrasterizer 1.13

#### 1.1170.1 Available under license:

No license file was found, but licenses were detected in source scan.

```plaintext
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#     http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-
  jar/org/apache/batik/apps/rasterizer/resources/Messages.properties

No license file was found, but licenses were detected in source scan.

// ----------------------------------------------------------------------------
// Licensed to the Apache Software Foundation (ASF) under one or more
// contributor license agreements. See the NOTICE file distributed with
// this work for additional information regarding copyright ownership.
// The ASF licenses this file to You under the Apache License, Version 2.0
// (the "License"); you may not use this file except in compliance with
// the License. You may obtain a copy of the License at
//     http://www.apache.org/licenses/LICENSE-2.0
// Unless required by applicable law or agreed to in writing, software
// distributed under the License is distributed on an "AS IS" BASIS,
// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```
grant codeBase "${app.dev.base}/classes/" {
   permission java.security.AllPermission;
};

grant codeBase "${app.dev.base}/lib/xml-apis-dom3.jar" {
   permission java.security.AllPermission;
};

grant codeBase "${app.dev.base}/lib/xerces_2_5_0.jar" {
   permission java.security.AllPermission;
};

grant codeBase "${app.dev.base}/lib/fop-transcoder-allinone-${xmlgraphics.fop.version}.jar" {
   permission java.security.AllPermission;
};

grant codeBase "${app.dev.base}/lib/xmlgraphics-commons-${xmlgraphics.commons.version}.jar" {
   permission java.security.AllPermission;
};

grant codeBase "${app.dev.base}/lib/js.jar" {
   permission java.lang.RuntimePermission "createClassLoader";
   permission java.net.SocketPermission "*", "listen, connect, resolve, accept";
   permission java.lang.RuntimePermission "accessDeclaredMembers";
   permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapter", "read";
   permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapterClassName", "read";
   permission java.io.FilePermission "<<ALL FILES>>", "read";
};

grant {
   permission java.io.FilePermission "resources/org/apache/batik/dom/svg/resources/svg10.dtd", "read";
   permission java.io.FilePermission "resources/org/apache/batik/dom/svg/resources/UserAgentStyleSheet.css", "read";
};

Found in path(s):
  * /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/resources/rasterizer.policy
No license file was found, but licenses were detected in source scan.
grant codeBase "${app.jar.base}/lib/xml-apis-dom3.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/xerces_2_5_0.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/batik-rasterizer.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-ext.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-dom.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-css.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-svg-dom.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-gvt.jar" {
    permission java.security.AllPermission;
};
grant codeBase "${app.jar.base}/lib/batik-parser.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-script.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-bridge.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-swing.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-transcoder.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/pdf-transcoder.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-gui-util.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-awt-util.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-util.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/batik-xml.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/xmlgraphics-commons-${xmlgraphics.commons.version}" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/lib/js.jar" {
    permission java.lang.RuntimePermission "createClassLoader";
};
permission java.net.SocketPermission "+", "listen, connect, resolve, accept";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapter", "read";
permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapterClassName", "read";
permission java.io.FilePermission "<<ALL FILES>>", "read";
};

grant {
    permission java.io.FilePermission "lib/batik-svg-dom.jar", "read";
};

Found in path(s):
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-
  jar/org/apache/batik/apps/rasterizer/resources/rasterizer.bin.policy
No license file was found, but licenses were detected in source scan.

// ----------------------------------------------------------------------------
// // Licensed to the Apache Software Foundation (ASF) under one or more
// // contributor license agreements. See the NOTICE file distributed with
// // this work for additional information regarding copyright ownership.
// // The ASF licenses this file to You under the Apache License, Version 2.0
// // (the "License"); you may not use this file except in compliance with
// // the License. You may obtain a copy of the License at
// // http://www.apache.org/licenses/LICENSE-2.0
// // Unless required by applicable law or agreed to in writing, software
// // distributed under the License is distributed on an "AS IS" BASIS,
// // WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
// // See the License for the specific language governing permissions and
// // limitations under the License.
// // $Id: rasterizer.policy.ref 1733416 2016-03-03 07:07:13Z gadams $
// ----------------------------------------------------------------------------

grant codeBase "+\{app.jar.base\}/@jarsLibDir@xml-apis-ext-1.3.04.jar" {
    permission java.security.AllPermission;
};

grant codeBase "+\{app.jar.base\}/@jarsLibDir@xml-apis-1.3.04.jar" {
    permission java.security.AllPermission;
};

grant codeBase "+\{app.jar.base\}/@jarsLibDir@xalan-2.7.0.jar" {
    permission java.security.AllPermission;
};
grant codeBase "${app.jar.base}/@jarsLibDir@xerces_2_5_0.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-rasterizer-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-ext-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-dom-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-css-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-svg-dom-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-gvt-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-parser-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-script-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-bridge-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-swing-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-transcoder-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-gui-util-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-awt-util-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-util-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-xml-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@fop-transcoder-allinone-@fopVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@xmlgraphics-commons-@xgcVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-anim-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-codec-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-extension-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@batik-svggen-@batikVersion@.jar" {
    permission java.security.AllPermission;
};

grant codeBase "${app.jar.base}/@jarsLibDir@js.jar" {
    permission java.lang.RuntimePermission "createClassLoader";
    permission java.net.SocketPermission "+", "listen, connect, resolve, accept";
    permission java.lang.RuntimePermission "accessDeclaredMembers";
    permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapter", "read";
    permission java.util.PropertyPermission "org.mozilla.javascript.JavaAdapterClassName", "read";
    permission java.io.FilePermission "<<ALL FILES>>", "read";
};
grant {
  permission java.io.FilePermission "${app.jar.base}/@jarsLibDir@batik-svg-dom-@batikVersion@.jar", "read";
};

Found in path(s):
  * /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/resources/rasterizer.policy.ref
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership.

The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership.

The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/
Found in path(s):
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/Main.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverter.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/DestinationType.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverterException.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverterController.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverterFileSource.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/DefaultSVGConverterController.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverterSource.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/Messages.java
* /opt/cola/permits/1146690669_1616599310.58/0/batik-svgrasterizer-1-13-sources-1-jar/org/apache/batik/apps/rasterizer/SVGConverterURLSource.java

1.1171 hive-shims 3.1.2
1.1171.1 Available under license:

Hive Shims
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1172 keypress 0.1.0 0.1.0

1.1172.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1173 hadoop-yarn-client 3.2.2

1.1173.1 Available under license:

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveledb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4, which has the following notices:

The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fastxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
  Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
  TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,

which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/ (Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUntime Library Exception"

== Contributors ==
* Tatu Saloranta
* Providing benchmark suite
* Alec Wysoker
* Performance and memory usage improvement

The binary distribution of this product bundles binaries of
Xerces2 Java Parser 2.9.1,
which has the following notices:
* =========================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License, ==
==  Version 2.0, in this case for the Apache Xerces Java distribution. ==
=========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation
This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* =----------------------------------------------------------------------------------------------------------------------------------------
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  =----------------------------------------------------------------------------------------------------------------------------------------

This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64,
LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 */
* Copyright (c) 2005, European Commission project OneLab under contract
* 034819 (http://www.one-lab.org)
* All rights reserved.
* Redistribution and use in source and binary forms, with or
* without modification, are permitted provided that the following
* conditions are met:
* - Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* - Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the distribution.
* - Neither the name of the University Catholique de Louvain - UCL
  nor the names of its contributors may be used to endorse or
  promote products derived from this software without specific prior
  written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
  "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
  LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
  FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
  COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
  INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
  BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
  LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
  CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
  LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
  ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
  POSSIBILITY OF SUCH DAMAGE.
*/

For portions of the native implementation of slicing-by-8 CRC calculation
in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor
the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 *
 * Abstract: The main routine
 *
 *--*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/\{lz4.h,lz4.c,lz4hc.h,lz4hc.c\},

/*
 LZ4 - Fast LZ compression algorithm
 Header File
 Copyright (C) 2011-2014, Yann Collet.
 BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

 * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

*/

For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

---------------------------------------------------------------------
(GO license)
This is a work derived from Russ Cox’s RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its contributors
  may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h
---------------------------------------------------------------------

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-
nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from
the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

=============

The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following
license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
  hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/1.10.7/

--------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

--------------------------------------------------------------------------------
For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

--------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
- bootstrap v3.3.6
- broccoli-asset-rev v2.4.2
- broccoli-funnel v1.0.1
- datatables v1.10.8
- em-helpers v0.5.13
- em-table v0.1.6
- ember v2.2.0
- ember-array-contains-helper v1.0.2
- ember-bootstrap v0.5.1
- ember-cli v1.13.13
- ember-cli-app-version v1.0.0
- ember-cli-babel v5.1.6
- ember-cli-content-security-policy v0.4.0
- ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
--------------------------------------------------------------------------------
Copyright (c) 2011-2014 Novus Partners, Inc.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

--------------------------------------------------------------------------------
The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

--------------------------------------------------------------------------------
MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

For:
oj! Algorithms - version 43.0
--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js
--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js
--------------------------------------------------------------------------------

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQLDB Database 2.3.4
(HSQL License)

"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1.Definitions.

1.1.Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2.Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3.Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.
2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a)rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b)otherwise make it clear that the
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF
COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY
PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY
CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF
GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER
COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE
POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO
LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO
THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT
ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO
THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in
48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as
that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer
software documentation as such terms are used in 48C.F.R.12.212 (Sept.
227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software
with only those rights set forth herein. This U.S. Government Rights clause is
in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision
that addresses Government rights in computer software under this
License.

9. MISCELLANEOUS.

This License represents the complete agreement
concerning subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to
make it enforceable. This License shall be governed by the law of the
jurisdiction specified in a notice contained within the Original Software
(except to the extent applicable law, if any, provides otherwise), excluding
such jurisdictions conflict-of-law provisions. Any litigation relating to this
License shall be subject to the jurisdiction of the courts located in the
jurisdiction and venue specified in a notice contained within the Original
Software, with the losing party responsible for costs, including, without
limitation, court costs and reasonable attorneys fees and expenses. The
application of the United Nations Convention on Contracts for the International
Sale of Goods is expressly excluded. Any law or regulation which provides that
the language of a contract shall be construed against the drafter shall not
apply to this License. You agree that You alone are responsible for compliance
with the United States export administration regulations (and the export control
laws and regulation of any other countries) when You use, distribute or
otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is
responsible for claims and damages arising, directly or indirectly, out of its
utilization of rights under this License and You agree to work with Initial
Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
gizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.
3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed
or otherwise made available under any subsequent version of the License, You
must distribute and make the Covered Software available under the terms of the
version of the License under which You originally received the Covered Software.
Otherwise, You may also choose to use, distribute or otherwise make the Covered
Software available under the terms of any subsequent version of the License
published by the license steward.
4.3. Modified Versions.
When You are an
Initial Developer and You want to create a new license for Your Original
Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward
(except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS
LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED
SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY
RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF
WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED
SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will
terminate automatically if You fail to comply with terms herein and fail to cure
such breach within 30 days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination of this License shall
survive.
6.2. If You assert a patent infringement claim (excluding declaratory
judgment actions) against Initial Developer or a Contributor (the Initial
Developer or Contributor against whom You assert such claim is referred to as
Participant) allegling that the Participant Software (meaning the
Contributor Version where the Participant is a Contributor or the Original
Software where the Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or indirectly to
You by such Participant, the Initial Developer (if the Initial Developer is not
the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this
License shall, upon 60 days notice from Participant terminate prospectively and
automatically at the expiration of such 60 day notice period, unless if within
such 60 day period You withdraw Your claim with respect to the Participant
Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.
4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the
intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:

JSch 0.1.54
Paranamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)
The WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work
will not be considered a Derivative Work for the purpose of this License. For
the avoidance of doubt, where the Work is a musical composition or sound
recording, the synchronization of the Work in timed-relation with a moving image
("synching") will be considered a Derivative Work for the purpose of this
License.

"Licensor" means the individual or entity that offers the Work under the terms
of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of
this License.

"You" means an individual or entity exercising rights under this License who has
not previously violated the terms of this License with respect to the Work, or
who has received express permission from the Licensor to exercise rights under
this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or
restrict any rights arising from fair use, first sale or other limitations on
the exclusive rights of the copyright owner under copyright law or other
applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor
hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the
duration of the applicable copyright) license to exercise the rights in the Work
as stated below:

to reproduce the Work, to incorporate the Work into one or more Collective
Works, and to reproduce the Work as incorporated in the Collective Works;
to create and reproduce Derivative Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission the Work including as
incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission Derivative Works.
For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive
right to collect, whether individually or via a performance rights society (e.g.
ASCAP, BMI, SESAC), royalties for the public performance or public digital
performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right
to collect, whether individually or via a music rights agency or designated
agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created
by 17 USC Section 115 of the US Copyright Act (or the equivalent in other
jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the
Work is a sound recording, Licensor waives the exclusive right to collect,
whether individually or via a performance-rights society (e.g. SoundExchange),
royalties for the public digital performance (e.g. webcast) of the Work, subject
to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).
The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer
UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the
party to be charged with such waiver or consent. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

--------------------------------------------------------------------------------
(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution means Covered Software of a particular Contributor.

1.4. Covered Software means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form means any form of the work other than Source Code Form.

1.7. Larger Work means a work that combines Covered Software with other material, in a separate
file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the
time of the initial grant or subsequently, any and all of the rights conveyed by
this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or
modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and
apparatus claims, in any patent Licensable by such Contributor that would be
infringed, but for the grant of the License, by the making, using, selling,
offering for sale, having made, import, or transfer of either its Contributions
or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General
Public License, Version 2.1, the GNU Affero General Public License, Version 3.0,
or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For
legal entities, You includes any entity that controls, is controlled by,
or is under common control with You. For purposes of this definition,
control means (a) the power, direct or indirect, to cause the direction or
management of such entity, whether by contract or otherwise, or (b) ownership of
more than fifty percent (50%) of the outstanding shares or beneficial ownership
of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:
under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.
2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).
3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous
This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

--------------------------------------------------------------------------------
/*--
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:
"This product includes software developed by the JDOM Project (http://www.jdom.org/)."
Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT}
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions
a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or
lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit
as required by Section 4(b), as requested. If You create an
Adaptation, upon notice from any Licensor You must, to the extent
practicable, remove from the Adaptation any credit as required by
Section 4(b), as requested.
b. If You Distribute, or Publicly Perform the Work or any Adaptations or
Collections, You must, unless a request has been made pursuant to
Section 4(a), keep intact all copyright notices for the Work and
provide, reasonable to the medium or means You are utilizing: (i) the
name of the Original Author (or pseudonym, if applicable) if supplied,
and/or if the Original Author and/or Licensor designate another party
or parties (e.g., a sponsor institute, publishing entity, journal) for
attribution ("Attribution Parties") in Licensor's copyright notice,
terms of service or by other reasonable means, the name of such party
or parties; (ii) the title of the Work if supplied; (iii) to the
extent reasonably practicable, the URL if any, that Licensor
specifies to be associated with the Work, unless such URL does not
refer to the copyright notice or licensing information for the Work;
and (iv), consistent with Section 3(b), in the case of an Adaptation,
a credit identifying the use of the Work in the Adaptation (e.g.,
"French translation of the Work by Original Author," or "Screenplay
based on original Work by Original Author"). The credit required by
this Section 4(b) may be implemented in any reasonable manner;
provided, however, that in the case of a Adaptation or Collection, at
a minimum such credit will appear, if a credit for all contributing
authors of the Adaptation or Collection appears, then as part of these
credits and in a manner at least as prominent as the credits for the
other contributing authors. For the avoidance of doubt, You may only
use the credit required by this Section for the purpose of attribution
in the manner set out above and, by exercising Your rights under this
License, You may not implicitly or explicitly assert or imply any
connection with, sponsorship or endorsement by the Original Author,
Licensor and/or Attribution Parties, as appropriate, of You or Your
use of the Work, without the separate, express prior written
permission of the Original Author, Licensor and/or Attribution
Parties.
c. Except as otherwise agreed in writing by the Licensor or as may be
otherwise permitted by applicable law, if You Reproduce, Distribute or
Publicly Perform the Work either by itself or as part of any
Adaptations or Collections, You must not distort, mutilate, modify or
take other derogatory action in relation to the Work which would be
prejudicial to the Original Author's honor or reputation. Licensor
agrees that in those jurisdictions (e.g. Japan), in which any exercise
of the right granted in Section 3(b) of this License (the right to
make Adaptations) would be deemed to be a distortion, mutilation,
modification or other derogatory action prejudicial to the Original
Author's honor and reputation, the Licensor will waive or not assert,
as appropriate, this Section, to the fullest extent permitted by the
applicable national law, to enable You to reasonably exercise Your
right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage.
guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

--------------------------------------------------------------------------------


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1174 isstream 0.1.2 0.1.2
1.1174.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Rod Vagg

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
1.1175 caseless 0.12.0 0.12.0

1.1175.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section)
patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
   You must give any other recipients of the Work or Derivative Works a copy of this License; and
   You must cause any modified files to carry prominent notices stating that You changed the files; and
   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1176 loader-utils 1.4.0 1.4.0

1.1176.1 Available under license:

Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1177 isarray 1.0.0 1.0.0

1.1177.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1.1178 color-convert 2.0.1

1.1178.1 Available under license:

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1179 scylla-driver-core 3.10.2-scylla-1

1.1179.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
No license file was found, but licenses were detected in source scan.

/*
 * Copyright DataStax, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/VIntCoding.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/UnsupportedProtocolVersionException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/AbstractReconnectionHandler.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/querybuilder/BindMarker.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/ClusteringOrder.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/AbstractData.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/schemabuilder/AlterKeyspace.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/InboundTrafficMeter.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/DirectedGraph.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/InvalidTypeException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/FunctionMetadata.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/GettableByNameData.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/TruncateException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/MD5Digest.java
Open Source Used In DNAC Platform Arica 4375

* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/querybuilder/Batch.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/FrameCompressor.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/Requests.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/ExtendedAuthProvider.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/TableMetadata.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/policies/RollingCount.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/SniEndPointFactory.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/SchemaChangeListenerBase.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/DataClassNameParser.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/DataType.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/exceptions/QueryValidationException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/schemabuilder/KeyspaceOptions.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/policies/RoundRobinPolicy.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/LZ4Compressor.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/DataTypeCqlNameParser.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/TypeTokens.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/SimpleStatement.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/ResultSet.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/PercentileTracker.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/MetricsUtil.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/UserType.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/ArrayBackedRow.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-
  jar/com/datastax/driver/core/SystemProperties.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/PreparedStatement.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/DriverInternalError.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/OverloadedException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/SyntaxError.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/MaterializedViewMetadata.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/CloudConfig.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/StreamIdGenerator.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/GettableData.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/querybuilder/Ordering.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/AuthProvider.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/QueryLogger.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/LatencyTracker.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/PagingStateException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/GettableByIndexData.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/CodecUtils.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/MessageToSegmentEncoder.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/policies/WhiteListPolicy.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/Session.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/SniEndPoint.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/NoHostAvailableException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/WriteTimeoutException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/schemabuilder/Create.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/exceptions/UnauthorizedException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/Responses.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/AbstractMonotonicTimestampGenerator.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/RemoteEndpointAwareNettySSLOptions.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/schemabuilder/DropKeyspace.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/schemabuilder/StatementStart.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/exceptions/UnpreparedException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/EventDebouncer.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/TokenRange.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/EndPoint.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/exceptions/AlreadyExistsException.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/schemabuilder/SettableByNameData.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/BatchStatement.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/DefaultPreparedStatement.java
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources-jar/com/datastax/driver/core/exceptions/QueryConsistencyException.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright DataStax, Inc.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Polynomial chosen from https://users.ece.cmu.edu/~koopman/crc/index.html, by Philip Koopman
* *
* <p>This webpage claims a copyright to Philip Koopman, which he licenses under the Creative
* Commons Attribution 4.0 International License (https://creativecommons.org/licenses/by/4.0)
It is unclear if this copyright can extend to a 'fact' such as this specific number, particularly as we do not use Koopman's notation to represent the polynomial, but we anyway attribute his work and link the terms of his license since they are not incompatible with our usage and we greatly appreciate his work.

This polynomial provides hamming distance of 8 for messages up to length 105 bits; we only support 8-64 bits at present, with an expected range of 40-48.

No license file was found, but licenses were detected in source scan.

Copyright DataStax, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

No license file was found, but licenses were detected in source scan.

Copyright (C) 2018 ScyllaDB

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/ShardingInfo.java
No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1202446501_1631567962.06/0/scylla-driver-core-3-10-2-scylla-1-sources.jar/com/datastax/driver/core/Driver.properties

1.1180 kerb-util 1.0.0
1.1180.1 Available under license :

Kerby-kerb Util
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1181 resourcelocator 1.0.3

1.1181.1 Available under license :

Found license 'Eclipse Public License 1.0' in *
Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights
reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public
License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'
Found license 'General Public License 2.0' in *
Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights
reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public
1.1182 hk2-utils 2.5.0-b32
1.1182.1 Available under license:
CDDL + GPLv2 with classpath exception
https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
A business-friendly OSS license

1.1183 murmur 1.0.0
1.1183.1 Available under license:
The Apache Software License, Version 2.0
http://www.apache.org/licenses/LICENSE-2.0.txt

1.1184 httpdlog-parser 5.7
1.1184.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Apache HTTPD & NGINX Access log parsing made easy
 * Copyright (C) 2011-2019 Niels Basjes
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-jar/nl/basjes/parse/httpdlog/NginxHttpdLogFormatDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-jar/nl/basjes/parse/httpdlog/HttpdLogFormatDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-jar/nl/basjes/parse/httpdlog/dissectors/HttpUriDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
jar/nl/basjes/parse/httpdlog/dissectors/ResponseSetCookieListDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/translate/ConvertNumberIntoCLF.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/tokenformat/Version.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/translate/StrTimeToDateTimeFormatter.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/HttpdLoglineParser.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/SslModule.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/RequestCookieListDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/GeoIPModule.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/UpstreamModule.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/tokenformat/NamedTokenParser.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/QueryStringFieldDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/ModUniqueIdDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/TokenOutputField.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/TokenParser.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/HttpExceptionDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/tokenformat/TokenFormatterDissector.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/tokenformat/TokenOutputField.java
* /opt/cola/permissions/1202446997_1631567998.12/0/httpdlog-parser-5.7-sources-jar/nl/basjes/parse/httpdlog/dissectors/ModUniqueIdDissector.java
Open Source Used In DNAC Platform Arica  4392

* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/ResponseSetCookieDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/KubernetesIngressModule.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/UpstreamListDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/HttpFirstLineDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/translate/ConvertMillisecondsIntoMicroseconds.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/CoreLogModule.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/nginxmodules/VariousModule.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPCountryDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPISPDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPCityDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/StrftimeStampDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/translate/ConvertCLFIntoNumber.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/translate/ConvertSecondsWithMillisStringDissector.java

No license file was found, but licenses were detected in source scan.

/*
* Apache HTTPD logparsing made easy
* Copyright (C) 2011-2019 Niels Basjes
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPCountryDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPISPDissector.java
* /opt/cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPCityDissector.java
* /opt/ cola/permits/1202446997_1631567998.12/0/httpdlog-parser-5-7-sources-
  jar/nl/basjes/parse/httpdlog/dissectors/geoip/GeoIPCityDissector.java
1.1185 unique-string 2.0.0

1.1185.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1186 define-property 0.2.5

1.1186.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1187 drill-format-image 1.19.0

1.1187.1 Available under license:

Drill: Contrib: Format: Image
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1188 @babel/code-frame 7.0.0

1.1188.1 Available under license:
MIT License

Copyright (c) 2014-2018 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1189 joi 6.10.1 6.10.1
1.1189.1 Available under license:

Copyright (c) 2012-2014, Walmart and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  * The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/joi/graphs/contributors

1.1190 parseurl 1.3.2 1.3.2
1.1190.1 Available under license:

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1191 hsql-db 1.8.0.10

1.1191.1 Available under license:
No license file was found, but licenses were detected in source scan.

(SqFile processor v. %{1})
Distribution is permitted under the terms of the HSQLDB license.
(c) 2004-2007 Blaine Simpson and the HSQLDB Development Group.

\q to Quit.
\? lists Special Commands.
:? lists Edit-Buffer/History commands.
*? lists PL commands.

SPECIAL Commands begin with '\' and execute when you hit ENTER.
EDIT-BUFFER / HISTORY Commands begin with ':' and execute when you hit ENTER.
COMMENTS begin with '/*' and end with the very next '*'.
PROCEDURAL LANGUAGE commands begin with '*' and end when you hit ENTER.
All other lines comprise SQL Statements.
SQL Statements are terminated by either a blank line (which moves the statement into the edit buffer without executing) or a line ending with ';'
(which executes the statement).
SQL Commands may begin with '/PLVARNAME' for a PL variable command alias.
After turning on variable expansion with command "*" (or any other PL command), PL variables may be used in any command (other than : commands) like so: *[PLVARNAME].

---

Open Source Used In DNAC Platform Arica 4400
1.1192 httpclient5 5.0.3

1.1192.1 Available under license:

Apache HttpClient
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1193 pkg-dir 2.0.0
1.1193.1 Available under license : MIT

1.1194 jline 2.14.6
1.1194.1 Available under license :

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
1.1195 parquet-avro 1.12.0

1.1195.1 Available under license:

Apache Parquet MR (Incubating)
Copyright 2014-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--------------------------------------------------------------------------------

This product includes code from Apache Avro, which includes the following in its NOTICE file:

Apache Avro
Copyright 2010-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, public display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product includes code from Apache Avro.

Copyright: 2014 The Apache Software Foundation.
Home page: https://avro.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

1.1196 dtrace-provider 0.6.0 0.6.0

1.1196.1 Available under license:

Copyright 2011 Chris Andrews. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY CHRIS ANDREWS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL CHRIS ANDREWS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2012 Chris Andrews. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

This software is provided by Chris Andrews "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall Chris Andrews or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

1.1197 map-stream 0.1.0 0.1.0

1.1197.1 Available under license:
Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
1.1198 cross-spawn 6.0.5 6.0.5

1.1198.1 Available under license:

   The MIT License (MIT)

   Copyright (c) 2018 Made With MOXY Lda <hello@moxy.studio>

   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
   all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
   THE SOFTWARE.

1.1199 metrics-jvm 4.1.0

1.1200 commons-digester 1.8.1

1.1200.1 Available under license:

   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Digester
Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.1201 ant-antlr 1.10.9
1.1201.1 Available under license :

/*
 * Apache License
 * Version 2.0, January 2004
 * http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 * *
 * "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
 *
 * "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
 *
 * "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation, source, and configuration files.
 *
 * "Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
Open Source Used In DNAC Platform Arica 4418

* and conversions to other media types.

* "Work" shall mean the work of authorship, whether in Source or
  Object form, made available under the License, as indicated by a
  copyright notice that is included in or attached to the work
  (an example is provided in the Appendix below).

* "Derivative Works" shall mean any work, whether in Source or Object
  form, that is based on (or derived from) the Work and for which the
  editorial revisions, annotations, elaborations, or other modifications
  represent, as a whole, an original work of authorship. For the purposes
  of this License, Derivative Works shall not include works that remain
  separable from, or merely link (or bind by name) to the interfaces of,
  the Work and Derivative Works thereof.

* "Contribution" shall mean any work of authorship, including
  the original version of the Work and any modifications or additions
  to that Work or Derivative Works thereof, that is intentionally
  submitted to Licensor for inclusion in the Work by the copyright owner
  or by an individual or Legal Entity authorized to submit on behalf of
  the copyright owner. For the purposes of this definition, "submitted"
  means any form of electronic, verbal, or written communication sent
  to the Licensor or its representatives, including but not limited to
  communication on electronic mailing lists, source code control systems,
  and issue tracking systems that are managed by, or on behalf of, the
  Licensor for the purpose of discussing and improving the Work, but
  excluding communication that is conspicuously marked or otherwise
  designated in writing by the copyright owner as "Not a Contribution."

* "Contributor" shall mean Licensor and any individual or Legal Entity
  on behalf of whom a Contribution has been received by Licensor and
  subsequently incorporated within the Work.

* 2. Grant of Copyright License. Subject to the terms and conditions of
  this License, each Contributor hereby grants to You a perpetual,
  worldwide, non-exclusive, no-charge, royalty-free, irrevocable
  copyright license to reproduce, prepare Derivative Works of,
  publicly display, publicly perform, sublicense, and distribute the
  Work and such Derivative Works in Source or Object form.

* 3. Grant of Patent License. Subject to the terms and conditions of
  this License, each Contributor hereby grants to You a perpetual,
  worldwide, non-exclusive, no-charge, royalty-free, irrevocable
  (except as stated in this section) patent license to make, have made,
  use, offer to sell, sell, import, and otherwise transfer the Work,
  where such license applies only to those patent claims licensable
  by such Contributor that are necessarily infringed by their
  Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

W3C SOFTWARE NOTICE AND LICENSE
https://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date
changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pdf

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
Apache Ant
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

1.1202 has-values 0.1.4 0.1.4
1.1202.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1203 calcite-linq4j 1.21.0-drill-r2
1.1203.1 Available under license :
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to you under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
# Database of contributors to Apache Calcite.
# Pages such as developer.md use this data.
# List must be sorted by first name, last name.
#
- name: Alan Gates
  emeritus: 2018/05/04
  apacheId: gates
  githubId: alanfgates
  org: Hortonworks
  role: PMC
- name: Aman Sinha
  apacheId: amansinha
  githubId: amansinha100
  org: MapR
  role: PMC
- name: Andrei Sereda
  apacheId: sereda
  githubId: asereda-gs
  org:
  role: Committer
- name: Ashutosh Chauhan
  apacheId: hashutosh
  githubId: ashutoshe
  org: Hortonworks
  role: PMC
- name: Chris Baynes
  apacheId: cbaynes
  githubId: chris-baynes
  org: Contiamo
  role: Committer
- name: Christian Beikov
  apacheId: cbeikov
  githubId: beikov
  org: Blazebit
  role: Committer
- name: Chunwei Lei
  apacheId: chunwei
  githubId: chunweilei
  org: Alibaba
  role: Committer
- name: Danny Chan
apacheId: danny0405
githubId: danny0405
org: Alibaba
role: Committer
- name: Edmon Begoli
  apacheId: ebegoli
githubId: ebegoli
org: Oak Ridge National Laboratory
role: Committer
- name: Francis Chuang
  apacheId: francischuang
githubId: F21
org: Boostport
role: PMC Chair
- name: Gian Merlino
  apacheId: gian
githubId: gianm
org: Imply
role: Committer
- name: Haisheng Yuan
  apacheId: hyuan
githubId: hsyuan
org: Alibaba
role: Committer
- name: Hongze Zhang
  apacheId: hongze
githubId: zhztheplayer
org: Tencent
role: Committer
- name: James R. Taylor
  apacheId: jamestaylor
githubId: JamesRTaylor
org: Salesforce
role: PMC
- name: Jacques Nadeau
  apacheId: jacques
githubId: jacques-n
org: Dremio
role: PMC
- name: Jess Balint
  apacheId: jbalint
githubId: jbalint
org: Stardog
role: Committer
- name: Jess Camacho Rodriguez
  apacheId: jcamacho
githubId: jcamachor
org: Hortonworks
role: PMC
- name: Jinfeng Ni
  apacheId: jni
  githubId: jinfengni
  org: MapR
  role: PMC
- name: John Pullokkaran
  apacheId: jpullokk
  githubId: jpullokkaran
  org: Hortonworks
  role: PMC
- name: Josh Elser
  apacheId: elserj
  githubId: joshelser
  org: Hortonworks
  role: PMC
- name: Julian Hyde
  apacheId: jhyde
  githubId: julianhyde
  org: Looker
  role: PMC
  homepage: http://people.apache.org/~jhyde
- name: Kevin Liew
  apacheId: kliew
  githubId: kliewkliew
  org:
  role: Committer
- name: Kevin Risden
  apacheId: krisden
  githubId: risdenk
  org:
  role: PMC
- name: Laurent Goujon
  apacheId: laurent
  githubId: laurentgo
  org: Dremio
  role: PMC
- name: Maryann Xue
  apacheId: maryannxue
  githubId: maryannxue
  org: Intel
  role: PMC
- name: Michael Mior
  apacheId: mmior
  githubId: michaelmior
  org: Rochester Institute of Technology
  role: PMC
  homepage: https://michael.mior.ca/
- name: Milinda Pathirage
  apacheId: milinda
  githubId: milinda
  org: Indiana University
  role: Committer
  homepage: http://milinda.pathirage.org/
- name: MinJi Kim
  apacheId: minji
  githubId: minji-kim
  org: Oracle
  role: Committer
- name: Nick Dimiduk
  apacheId: ndimiduk
  githubId: ndimiduk
  org: Hortonworks
  role: PMC
  homepage: http://www.n10k.com
- name: Nishant Bangarwa
  apacheId: nishant
  githubId: nishantmonu51
  org: Hortonworks
  role: Committer
- name: Ruben Quesada Lopez
  apacheId: rubenql
  githubId: rubenada
  org: TIBCO
  role: Committer
- name: Sergey Nuyanzin
  apacheId: snuyanzin
  githubId: snuyanzin
  org: EPAM
  role: Committer
- name: Shuyi Chen
  apacheId: shuyichen
  githubId: suez1224
  org: Uber
  role: Committer
- name: Slim Bouguerra
  apacheId: bslim
  githubId: b-slim
  org: Hortonworks
  role: Committer
- name: Stamatis Zampetakis
  apacheId: zabetak
  githubId: zabetak
  org: TIBCO
  role: PMC
- name: Steven Noels
apacheId: stevenn
githubId: stevenn
org: NGData
role: PMC
- name: Ted Dunning
  apacheId: tdunning
githubId: tdunning
org: MapR
role: PMC
  avatar: https://mapr.com/blog/author/ted-dunning/assets/tdunning-panama.jpg
- name: Vladimir Sitnikov
  apacheId: vladimirsitnikov
githubId: vlsi
org: NetCracker
role: PMC
- name: Volodymyr Vysotskyi
  apacheId: volodymyr
githubId: vvysotskyi
org:
role: PMC
- name: Zhen Wang
  apacheId: zhenw
githubId: zinking
org:
role: Committer
- name: Zhiqiang He
  apacheId: zhiqianghe
githubId: Zhiqiang-He
org: Huawei
role: Committer
- name: Zhiwei Peng
  apacheId: zhiwei
githubId: pengzhiwei2018
org: Ant Financial Group
role: Committer
- name: Zoltan Haindrich
  apacheId: kgyrtkirk
githubId: kgyrtkirk
org: Hortonworks
role: Committer

# End contributors.yml

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent...
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------

APACHE CALCITE SUBCOMPONENTS:

The Apache Calcite project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

-----------------------------------------------------------------------

The MIT License

The Apache Calcite project bundles the following files under the MIT License:

- site
  Parts of the web site generated by Jekyll (http://jekyllrb.com/)
  Copyright (c) 2008-2015 Tom Preston-Werner
- site/_sass/_font-awesome.scss
  Font-awesome css files v4.1.0 (http://fortawesome.github.io/Font-Awesome/)
  Copyright (c) 2013 Dave Gandy
- site/_sass/_normalize.scss
  normalize.css v3.0.2 | git.io/normalize
  Copyright (c) Nicolas Gallagher and Jonathan Neal
- site/_sass/_gridism.scss
  Gridism: A simple, responsive, and handy CSS grid by @cobyism
  https://github.com/cobyism/gridism
  Copyright (c) 2013 Coby Chapple
- site/js/html5shiv.min.js
  HTML5 Shiv 3.7.2 | @afarkas @dalton @jon_neal @rem
- site/js/respond.min.js
  Respond.js v1.4.2: min/max-width media query polyfill
  Copyright 2013 Scott Jehl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------
The Open Font License
-----------------------------------------------------------------------

The Apache Calcite project bundles the following fonts under the SIL Open Font License (OFL) - http://scripts.sil.org/OFL/

- site/fonts/fontawesome-webfont.*
  Font-awesome font files v4.0.3 (http://fortawesome.github.io/Font-Awesome/)

Apache Calcite
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.

The web site includes files generated by Jekyll.

1.1204 react-is 16.12.0 16.12.0
1.1204.1 Available under license :
  MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1205 json-schema-traverse 0.4.1 0.4.1
1.1205.1 Available under license :
MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1206 jline-terminal-jna 3.12.1
1.1206.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2016, the original author or authors.
 * This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/solaris/SolarisNativePty.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/JnaNativePty.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/win/WindowsAnsWriter.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/osx/OsXNativePty.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/freebsd/CLibrary.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/osx/CLibrary.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/linux/CLibrary.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/solaris/CLibrary.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/freebsd/FreeBsdNativePty.java
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/linux/LinuxNativePty.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (c) 2002-2017, the original author or authors.
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/win/JnaWinConsoleWriter.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (c) 2002-2018, the original author or authors.
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/win/Kernel32.java
No license file was found, but licenses were detected in source scan.

2016, the original author or authors.

This software is distributable under the BSD license. See the terms of the
BSD license in the documentation provided with this software.

https://opensource.org/licenses/bsd

Found in path(s):
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/META-INF/maven/org.jline/jline-terminal-jna/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2019, the original author or authors.
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600364_1607534022.33/0/jline-terminal-jna-3-12-1-sources-jar/org/jline/terminal/impl/jna/win/JnaWinSysTerminal.java

1.1207 groovy-swing 3.0.8
1.1207.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/codehaus/groovy/binding/PropertyBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/model/NestedValueModel.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/TableModelFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/model/PropertyModel.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/ClosureSourceBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/FullBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/codehaus/groovy/binding/BindPath.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/PropertyBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/JTableMetaMethods.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/model/FormModel.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/AbstractFullBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/JComponentProperties.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/AbstractButtonProperties.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/TriggerBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/DialogFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/BindFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/codehaus/groovy/runtime/SwingGroovyMethods.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/WidgetFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/model/ClosureModel.java

Open Source Used In DNAC Platform Arica  4438
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/GridBagFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/RichActionWidgetFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/TextArgWidgetFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/WindowFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/JComboBoxProperties.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/SwingBorderFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/ComponentFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/codehaus/groovy/binding/AbstractFullBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/codehaus/groovy/binding/SwingTimerTriggerBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/TargetBinding.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/BoxFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/ButtonGroupFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/TabbedPaneFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/ComboBoxFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/ListFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/BeanFactory.groovy
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/DefaultTableModel.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/DefaultTableColumn.java
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/DefaultTableColumn.java
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/org/apache/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/model/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/impl/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/model/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/binding/package.html
* /opt/cola/permits/1218299960_1634924418.62/0/groovy-swing-3-0-8-sources-jar/groovy/swing/factory/package.html
1.1208 pdfbox 2.0.22
1.1208.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

EXTERNAL COMPONENTS

Apache PDFBox includes a number of components with separate copyright notices
and license terms. Your use of these components is subject to the terms and
conditions of the following licenses.

Contributions made to the original PDFBox project:

Copyright (c) 2002-2007, www.pdfbox.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of pdfbox; nor the names of its contributors may be
   used to endorse or promote products derived from this software without
   specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Adobe Font Metrics (AFM) for PDF Core 14 Fonts

This file and the 14 PostScript(R) AFM files it accompanies may be used, copied, and distributed for any purpose and without charge, with or without modification, provided that all copyright notices are retained; that the AFM files are not distributed without this file; that all modifications to this file or any of the AFM files are prominently noted in the modified file(s); and that this paragraph is not modified. Adobe Systems has no responsibility or obligation to support the use of the AFM files.

CMaps for PDF Fonts (http://opensource.adobe.com/wiki/display/cmap/Downloads)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this documentation file to use, copy, publish, distribute, sublicense, and/or sell copies of the documentation, and to permit others to do the same, provided that:
- No modification, editing or other alteration of this document is allowed; and
- The above copyright notice and this permission notice shall be included in all copies of the documentation.

Permission is hereby granted, free of charge, to any person obtaining a copy of this documentation file, to create their own derivative works from the content of this document to use, copy, publish, distribute, sublicense, and/or sell the derivative works, and to permit others to do the same, provided that the derived work is not represented as being a copy or version of this document.

Adobe shall not be liable to any party for any loss of revenue or profit or for indirect, incidental, special, consequential, or other similar damages, whether based on tort (including without limitation negligence or strict liability), contract or other legal or equitable grounds even if Adobe has been advised or had reason to know of the possibility of such damages. The Adobe materials are provided on an "AS IS" basis. Adobe specifically disclaims all express, statutory, or implied warranties relating to the Adobe materials, including but not limited to those concerning merchantability or fitness for a particular purpose or non-infringement of any third party rights regarding the Adobe materials.

Liberation Fonts (https://fedorahosted.org/liberation-fonts)

Digitized data copyright (c) 2010 Google Corporation
with Reserved Font Arimo, Tinos and Cousine.
Copyright (c) 2012 Red Hat, Inc.
with Reserved Font Name Liberation.

This Font Software is licensed under the SIL Open Font License, Version 1.1.

This license is copied below, and is also available with a FAQ at:
http://scripts.sil.org/OFL

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

PREAMBLE The goals of the Open Font License (OFL) are to stimulate
worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting in part or in whole any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS

Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or
binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Twelvemonkeys (https://github.com/haraldk/TwelveMonkeys/)

Copyright (c) 2008-2016, Harald Kuhr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name "TwelveMonkeys" nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

Apache PDFBox
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Based on source code contributed to the original PDFBox project.
Copyright (c) 2002-2007, www.pdfbox.org

Includes the Adobe Glyph List

Includes the Zapf Dingbats Glyph List

Includes the Bidi Mirroring Glyph Property (BidiMirroring-8.0.0.txt)
Copyright 1991-2015 Unicode, Inc.

Includes parts of TwelveMonkeys ImageIO
Copyright 2008-2016 Harald Kuhr

1.1209 bops 0.1.1 0.1.1
1.1209.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1210 jsonpath-plus 3.0.0 3.0.0

1.1210.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011-2019 Stefan Goessner, Subbu Allamaraju, Mike Brevoort, Robert Krahn, Brett Zamir, Richard Schneider

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1211 xpp 1.1.4c

1.1211.1 Notifications :

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

1.1211.2 Available under license :

/*
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2001-2003 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Axis" and "Apache Software Foundation" must
 *    not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache",
 *    nor may "Apache" appear in their name, without prior written
 *    permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF

Open Source Used In DNAC Platform Arica 4455
Open Source Used In DNAC Platform Arica 4456

* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* ====================================================================
* * This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* */

1.1212 asm 5.0.3
1.1212.1 Available under license:

No license file was found, but licenses were detected in source scan.

/****
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2011 INRIA, France Telecom
* All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* *
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/
/**
 * Creates a new [@link AnalyzerAdapter]. <i>Subclasses must not use this
 * constructor</i>. Instead, they must use the
 * [@link #AnalyzerAdapter(int, String, int, String, String, MethodVisitor)]
 * version.
 *
 * @param owner
 * the owner's class name.
 *
 * @param access
 * the method's access flags (see [@link Opcodes]).
 *
 * @param name
 * the method’s name.
 *
 * @param desc
 * the method's descriptor (see [@link Type Type]).
 *
 * @param mv
 * the method visitor to which this adapter delegates calls. May
 * be <tt>null</tt>.
 *
 * @throws IllegalStateException
 * If a subclass calls this constructor.
 */

Found in path(s):
/opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/AnalyzerAdapter.java

No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2011 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/
/**
 * Creates a new [@link GeneratorAdapter]. <i>Subclasses must not use this
 * constructor</i>. Instead, they must use the
 * [@link #GeneratorAdapter(int, MethodVisitor, int, String, String)]
 * version.
 *
 * [@param mv
 * the method visitor to which this adapter delegates calls.
 * [@param access
 * the method's access flags (see [@link Opcodes]).
 * [@param name
 * the method's name.
 * [@param desc
 * the method's descriptor (see [@link Type Type]).
 * @throws IllegalArgumentException
 * If a subclass calls this constructor.
 */

No license file was found, but licenses were detected in source scan.

/***
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2013 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/TypePath.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/TypeReference.java

No license file was found, but licenses were detected in source scan.

/****
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

/**
* Creates a new JSRInliner. <i>Subclasses must not use this
* constructor</i>. Instead, they must use the
* { @link #JSRInlinerAdapter(int, MethodVisitor, int, String, String, String, String[])}
* version.
*
* @param mv
* the <code>MethodVisitor</code> to send the resulting inlined
* method code to (use <code>null</code> for none).
* @param access
* the method's access flags (see { @link Opcodes}). This
* parameter also indicates if the method is synthetic and/or
* deprecated.
* @param name
* the method's name.
* @param desc
* the method's descriptor (see { @link Type}).
* @param signature
* the method's signature. May be <tt>null</tt>.
* @param exceptions
* the internal names of the method's exception classes (see
* { @link Type#getInternalName() getInternalName}). May be
* <tt>null</tt>.
* @throws IllegalStateException
* If a subclass calls this constructor.
*/

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2011 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/
/**
 * Constructs a new {@link LocalVariableAnnotationNode}.<i>Subclasses must
 * not use this constructor</i>. Instead, they must use the
 * {@link #LocalVariableAnnotationNode(int, TypePath, LabelNode[], LabelNode[], int[], String)}
 * version.
 *
 * @param typeRef
 *             a reference to the annotated type. See {@link TypeReference}.
 * @param typePath
 *             the path to the annotated type argument, wildcard bound, array
 *             element type, or static inner type within 'typeRef'. May be
 *             <tt>null</tt> if the annotation targets 'typeRef' as a whole.
 * @param start
 *             the fist instructions corresponding to the continuous ranges
 *             that make the scope of this local variable (inclusive).
 * @param end
 *             the last instructions corresponding to the continuous ranges
 *             that make the scope of this local variable (exclusive). This
 *             array must have the same size as the 'start' array.
 * @param index
 *             the local variable's index in each range. This array must have
 *             the same size as the 'start' array.
 * @param desc
 *             the class descriptor of the annotation class.
 */

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-
  jar/org/objectweb/asm/tree/LocalVariableAnnotationNode.java
No license file was found, but licenses were detected in source scan.

2011, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/asm-xml.dtd
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
**asm-5-0-3-sources-7-jar/org/objectweb/asm/util/package.html**

**asm-5-0-3-sources-7-jar/org/objectweb/asm/package.html**

**asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/package.html**

**asm-5-0-3-sources-7-jar/org/objectweb/asm/signature/package.html**

**asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/analysis/package.html**

**asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/package.html**

No license file was found, but licenses were detected in source scan.

```/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2011 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */
/**
 * Constructs a new [@link Textifier]. <i>Subclasses must not use this
 * constructor</i>. Instead, they must use the [@link #Textifier(int)]
* version.
* 
* @throws IllegalStateException
*         If a subclass calls this constructor.
*/

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/util/Textifier.java
No license file was found, but licenses were detected in source scan.

/****
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2011 INRIA, France Telecom
* All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/
/**
* Constructs a new [ClassNode](ClassNode). Subclasses must not use this
* constructor. Instead, they must use the [ClassNode(int)]
* version.
*
* @throws IllegalStateException
*         If a subclass calls this constructor.
*/
Open Source Used In DNAC Platform Arica 4465

No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2011 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */

No license file was found, but licenses were detected in source scan.

2011, Eugene Kuleshov
* All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/package.html
No license file was found, but licenses were detected in source scan.

/****
* ASM XML Adapter
* Copyright (c) 2004-2011, Eugene Kuleshov
* All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/Processor.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/SAXClassAdapter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/SAXAdapter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/SAXCodeAdapter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/ASMContentHandler.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/SAXFieldAdapter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/xml/SAXAnnotationAdapter.java

No license file was found, but licenses were detected in source scan.

/**
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
/*

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/ClassOptimizer.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/Label.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/Item.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/StaticInitMerger.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/RemappingMethodAdapter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/MethodVisitor.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/ClassVisitor.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/LocalVariablesSorter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/util/TraceMethodVisitor.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/commons/Method.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/MethodNode.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/LocalVariableNode.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-
No license file was found, but licenses were detected in source scan.

#All rights reserved.
#Redistribution and use in source and binary forms, with or without
#modification, are permitted provided that the following conditions
#are met:
#1. Redistributions of source code must retain the above copyright
   # notice, this list of conditions and the following disclaimer.
#2. Redistributions in binary form must reproduce the above copyright
   # notice, this list of conditions and the following disclaimer in the
   # documentation and/or other materials provided with the distribution.
#3. Neither the name of the copyright holders nor the names of its
# this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink.properties
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink-writer.properties
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink-resize.properties
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink-signatures.properties
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink-annotations.properties
* /opt/cola/permits/1135840046_1613613019.19/0/asm-5-0-3-sources-7-jar/org/objectweb/asm/optimizer/shrink-frames.properties

1.1213 big.js 5.2.2 5.2.2
1.1213.1 Available under license :

The MIT Licence (Expat).

Copyright (c) 2018 Michael Mclaughlin

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1214 type-fest 0.20.2
1.1214.1 Available under license:
Creative Commons Legal Code

CC0 1.0 Universal

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS DOCUMENT DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
ii. moral rights retained by the original author(s) and/or performer(s);
iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;
v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and
vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder
of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1215 arraybuffer.slice 0.0.7 0.0.7

1.1215.1 Available under license:

Copyright (C) 2013 Rase-

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1216 json-schema-ref-parser 6.1.0 6.1.0

1.1216.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 James Messinger

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1217 async 0.9.0 0.9.0

1.1217.1 Available under license:

Copyright (c) 2010-2014 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1218 babel-preset-env 1.7.0 1.7.0

1.1218.1 Available under license:

MIT License

Copyright (c) 2016-2017 Babel

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1219 component-emitter 1.3.0 1.3.0
1.1219.1 Available under license:
(The MIT License)

Copyright (c) 2014 Component contributors <dev@component.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1220 curator-framework 5.1.0
1.1220.1 Available under license:

Curator Framework
Copyright 2011-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1221 hbase-asyndfs 2.4.2

1.1221.1 Available under license:

Apache HBase - Asynchronous File System
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1222 jetty-security 9.4.41.v20210516

1.1222.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution ‘originates’
from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free copyright license to
   reproduce, prepare derivative works of, publicly display, publicly
   perform, distribute and sublicense the Contribution of such Contributor,
   if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free patent license under
   Licensed Patents to make, use, sell, offer to sell, import and otherwise
   transfer the Contribution of such Contributor, if any, in source code and
   object code form. This patent license shall apply to the combination of
   the Contribution and the Program if, at the time the Contribution is
   added by the Contributor, such addition of the Contribution causes such
   combination to be covered by the Licensed Patents. The patent license
   shall not apply to any other combinations which include the Contribution.
   No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other
intellectual property rights of any other entity. Each Contributor
disclaims any liability to Recipient for claims brought by any other
entity based on infringement of intellectual property rights or
otherwise. As a condition to exercising the rights and licenses granted
hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and

b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle
The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.
org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1223 groovy-xml 3.0.8
1.1223.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/package.html
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/package.html
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/package.html
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/dom/package.html
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/util/slurpersupport/package.html

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/namespace.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/FactorySupport.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/StandardXmlFilter.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/Attributes.java
Open Source Used In DNAC Platform Arica 4500

* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/DoubleQuoteFilter.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/Entity.groovy
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NamespaceAwareHashMap.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/FilteredNodeChildren.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/StandardControlToUndefined.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/XmlUtil.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/Attributes.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NodeChild.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/Node.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/org/apache/groovy/xml/tools/DomToGroovy.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/AllControlToUndefined.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/dom/DOMCategory.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/StreamingMarkupWriter.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/StreamingMarkupBuilder.groovy
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/BaseMarkupBuilder.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/markupsupport/SingleQuoteFilter.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/XmlParser.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NodeIterator.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NoChildren.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/StreamingMarkupBuilder.groovy
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/StreamingMarkupBuilder.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/Attributes.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NodeChild.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/Node.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NodeChildren.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NoChildren.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingmarkupsupport/StreamingMarkupWriter.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/dom/DOMCategory.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/extensions/XmlExtensions.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/slurpersupport/NodeIterator.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/streamingDOMBuilder.groovy
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/DOMBuilder.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/xml/DOMBuilder.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/util/XmlUtil.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/util/XmlUtil.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/util/XmlUtil.java
* /opt/cola/permits/1218302256_1634930914.09/0/groovy-xml-3-0-8-sources-jar/groovy/util/XmlParser.java
1.1224 @babel/parser 7.11.5
1.1224.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1225 expand-brackets 2.1.4 2.1.4
1.1225.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1226 metrics-health-checks 4.0.2

1.1227 tomcat-annotations-api 8.5.46

1.1227.1 Available under license:

Apache Tomcat
Copyright 1999-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Open Source Used In DNAC Platform Arica 4505

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

Work and such Derivative Works in Source or Object form.
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1228 hadoop-client 3.2.2
1.1228.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:

* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com
The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of
Gson 2.2.4,
which has the following notices:

    The Netty Project
    =================

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the
components that this product depends on.

-----------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has
been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jboscach experimental/jsr166/
This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java
This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of ‘Apache Commons Lang’, a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of ‘JDOM 1.1’, which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of
Commons Codec 1.4,
which has the following notices:
* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of
Commons Lang 2.6,
which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see:
  StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of
Apache Log4j 1.2.17,
which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:
The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
* ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception. 
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions. 
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

MortBay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* =========================================================================
  == NOTICE file corresponding to section 4(d) of the Apache License, ==
  == in this case for the Apache Xerces Java distribution. ==
=========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
* voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

The binary distribution of this product bundles binaries of Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:

* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

  This product includes software developed at
  The Apache Software Foundation (http://www.apache.org/).

  This product incorporates portions of the 'Hadoop' project
  Copyright 2007-2009 The Apache Software Foundation
  Licensed under the Apache License v2.0

  Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
  It is licensed Creative Commons Attribution 3.0.
  See https://creativecommons.org/licenses/by/3.0/us/
  We changed the logo by stripping the colored background, inverting
  it and then rotating it some.

  Later we found that vectorfree.com image is not properly licensed.
  The original is owned by vectorportal.com. The original was
  relicensed so we could use it as Creative Commons Attribution 3.0.
  The license is bundled with the download available here:

  This product includes portions of the Bootstrap project v3.0.0

  Copyright 2013 Twitter, Inc.

  Licensed under the Apache License v2.0

  This product uses the Glyphicons Halflings icon set.

  http://glyphicons.com/

  Copyright Jan Kovak

  Licensed under the Apache License v2.0 as a part of the Bootstrap project.

  This product includes portions of the Guava project v14, specifically
  'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

  Copyright (C) 2007 The Guava Authors
Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 2008-2015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 2007-2008 Robert Niederreiter, Jens Klein
# Copyright 2004-2005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
This product includes software developed by Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of Ehcache 3.3.1, which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of snakeml (https://bitbucket.org/asomov/snakeml), which has the following notices:
* Copyright (c) 2008, http://www.snakeml.org

The binary distribution of this product bundles binaries of swagger-annotations (https://github.com/swagger-api/swagger-core), which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of metrics-core 3.2.4, which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * - Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor
the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++

* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html
* Abstract: The main routine
* --*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/{lz4.h,lz4.c,lz4hc.h,lz4hc.c},

/*
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at :
- LZ4 source repository : http://code.google.com/p/lz4/
- LZ4 public forum : https://groups.google.com/forum/#!forum/lz4c

/*

For hadoop-common-project/hadoop-common/src/main/native/gtest
---------------------------------------------------------------

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
re2j 1.1

re2j 1.1

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
  distribution.

* Neither the name of Google Inc. nor the names of its contributors
  may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=============

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Open Source Used In DNAC Platform Arica 4530
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
--------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js
--------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
em-helpers v0.5.13
em-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.13.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinners v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25
----------------------------------------------------------------------------------------------------------------------------

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
/hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
--------------------------------------------------------------------------------

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this
file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the
License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
express or implied. See the License for the specific language governing permissions and
limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js
--------------------------------------------------------------------------------
The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery
Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1,jre7
MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

===

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:

hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js
--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js
--------------------------------------------------------------------------------
D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the
following license:
HSQldb Database 2.3.4

(HSQL License)
“COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQl Development Group:

Copyright (c) 2001-2016, The HSQl Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the HSQl Development Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL HSQl DEVELOPMENT GROUP, HSQLDB.ORG,
 OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
 PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group.

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0
1. Definitions.

1.1.Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2.Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered
Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities,
You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that
Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the
recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license.
4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to
termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control
laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
gizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions
thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software
3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or
new versions of this License from time to time. Each version will be given a
distinguishing version number. Except as provided in Section 4.3, no one other
than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the
Covered Software available under the terms of the version of the License under
which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed
or otherwise made available under any subsequent version of the License, You
must distribute and make the Covered Software available under the terms of the
version of the License under which You originally received the Covered Software.
Otherwise, You may also choose to use, distribute or otherwise make the Covered
Software available under the terms of any subsequent version of the License
published by the license steward.

4.3. Modified Versions.
When You are an
Initial Developer and You want to create a new license for Your Original
Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward
(except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS
LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED
SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY
RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF
WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED
SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will
terminate automatically if You fail to comply with terms herein and fail to cure
such breach within 30 days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination of this License shall
survive.

6.2. If You assert a patent infringement claim (excluding declaratory
judgment actions) against Initial Developer or a Contributor (the Initial
Developer or Contributor against whom You assert such claim is referred to as
Participant) alleging that the Participant Software (meaning the
Contributor Version where the Participant is a Contributor or the Original
Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of,
and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT’S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

“Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such
derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other combinations
which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to
its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to
Recipient for claims brought by any other entity based on infringement of
intellectual property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby assumes sole
responsibility to secure any other intellectual property rights needed, if any.
For example, if a third party patent license is required to allow Recipient to
distribute the Program, it is Recipient's responsibility to acquire that license
before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright
rights in its Contribution, if any, to grant the copyright license set forth in
this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its
own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and
conditions, express and implied, including warranties or conditions of title and
non-infringement, and implied warranties or conditions of merchantability and
fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and consequential
damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by
that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor,
and informs licensees how to obtain it in a reasonable manner on or through a
medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs
of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions)
under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
FindBugs-jsr305 3.0.0  
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a
Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats.

All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use
of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:
jamon-runtime 2.4.1

(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution means Covered Software of a particular Contributor.

1.4. Covered Software means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or
earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.
2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of...
Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered
Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a
particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a
jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If you choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.
If it is not possible or desirable to put the notice in a particular file, then
You may include the notice in a location (such as a LICENSE file in a relevant
directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined
by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the
following license:
JDOM 1.1

/*--
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions, and the disclaimer that follows
these conditions in the documentation and/or other materials
provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products
derived from this software without prior written permission. For
written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor
may "JDOM" appear in their name, without prior written permission
from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the
software itself an acknowledgement equivalent to the following:
"This product includes software developed by the
JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos
THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE
TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or
form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the
Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological
measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor
agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous
a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.
Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1229 babel-helper-hoist-variables 6.24.1

6.24.1

1.1229.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1230 is-descriptor 0.1.6 0.1.6
1.1230.1 Available under license:
   The MIT License (MIT)

   Copyright (c) 2015-2017, Jon Schlinkert.

   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
   all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
   THE SOFTWARE.

1.1231 picocli 4.5.2
1.1231.1 Available under license:
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.1232 protobuf-java 2.5.32

1.1232.1 Available under license:

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:
Sanjay Ghemawat <sanjay@google.com>
Jeff Dean <jeff@google.com>
Daniel Dulitz <daniel@google.com>
Craig Silverstein
Paul Haahr <haahr@google.com>
Corey Anderson <corin@google.com>
(and many others)

Proto2 C++ and Java primary author:
Kenton Varda <kenton@google.com>

Proto2 Python primary authors:
Will Robinson <robinson@google.com>
Petar Petrov <petar@google.com>

Java Nano primary authors:
Brian Duff <bduff@google.com>
Tom Chao <chaot@google.com>
Max Cai <maxtroy@google.com>
Ulas Kirazci <ulas@google.com>

Large code contributions:
Jason Hsueh <jasonh@google.com>
Joseph Schorr <jschorr@google.com>
Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:
Scott Bruce <sbruce@google.com>
Frank Yellin
Neal Norwitz <nnorwitz@google.com>
Jeffrey Yasskin <jyasskin@google.com>
Ambrose Feinstein <ambrose@google.com>

Documentation:
Lisa Carey <lcarey@google.com>

Maven packaging:
Gregory Kick <gak@google.com>

Patch contributors:
Kevin Ko <kevin.s.ko@gmail.com>
   * Small patch to handle trailing slashes in --proto_path flag.
Johan Euphrosine <proppy@aminche.com>
   * Small patch to fix Python CallMethod().
Ulrich Kunitz <kune@deine-taler.de>
   * Small optimizations to Python serialization.
Leandro Lucarella <lucax@gmail.com>
   * VI syntax highlighting tweaks.
   * Fix compiler to not make output executable.
Dilip Joseph <dilip.antony.joseph@gmail.com>
   * Heuristic detection of sub-messages when printing unknown fields in text format.
Brian Atkinson <nairb774@gmail.com>
   * Added @Override annotation to generated Java code where appropriate.
Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>
   * Tru64 support.
Monty Taylor <monty.taylor@gmail.com>
   * Solaris 10 + Sun Studio fixes.
Alek Storm <alek.storm@gmail.com>
   * Slicing support for repeated scalar fields for the Python API.
Oleg Smolsky <oleg.smolsky@gmail.com>
   * MS Visual Studio error format option.
   * Detect unordered_map in stl_hash.m4.
Brian Olson <brianolson@google.com>
   * gzip/zlib I/O support.
Michael Poole <mdpoole@troilus.org>
   * Fixed warnings about generated constructors not explicitly initializing all fields (only present with certain compiler settings).
   * Added generation of field number constants.
Wink Saville <wink@google.com>
   * Fixed initialization ordering problem in logging code.
Will Pierce <willp@nuclei.com>
   * Small patch improving performance of in Python serialization.
Alexandre Vassalotti <alexandre@peadrop.com>
   * Emacs mode for Protocol Buffers (editors/protobuf-mode.el).
Scott Stafford <scott.stafford@gmail.com>
   * Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.
Alexander Melnikov <alm@sibmail.ru>
   * HPUX support.
Oliver Jowett <oliver.jowett@gmail.com>
   * Detect whether zlib is new enough in configure script.
   * Fixes for Solaris 10 32/64-bit confusion.
Evan Jones <evanj@mit.edu>
* Optimize Java serialization code when writing a small message to a stream.
* Optimize Java serialization of strings so that UTF-8 encoding happens only once per string per serialization call.
* Clean up some Java warnings.
* Fix bug with permanent callbacks that delete themselves when run.

Michael Kucharski <m.kucharski@gmail.com>
* Added CodedInputStream.getTotalBytesRead().

Kacper Kowalik <xarthisius.kk@gmail.com>
* Fixed m4/acx_pthread.m4 problem for some Linux distributions.

William Orr <will@worrbase.com>
* Fixed detection of sched_yield on Solaris.
* Added atomicops for Solaris

Andrew Paprocki <andrew@ishiboo.com>
* Fixed minor IBM xIC compiler build issues
* Added atomicops for AIX (POWER)

# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Mocking Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Benoit Sigoure <tsuna@google.com>
Bogdan Piloca <boo@google.com>
Chandler Carruth <chandlerc@google.com>
Dave MacLachlan <dmaclach@gmail.com>
David Anderson <danderson@google.com>
Dean Sturtevant
Gene Volovich <gv@cite.com>
Hal Burch <gmock@hburch.com>
Jeffrey Yasskin <jyasskin@google.com>
Jim Keller <jimkeller@google.com>
Joe Walnes <joe@truemesh.com>
Jon Wray <jwray@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kostya Serebryany <kcc@google.com>
Lev Makhlis
Manuel Klimek <klimek@google.com>
Mario Tanev <radix@google.com>
Mark Paskin
Markus Heule <markus.heule@gmail.com>
Matthew Simmons <simmonmt@acm.org>
Mike Bland <mbland@gmail.com>
Neal Norwitz <nnorwitz@gmail.com>
Nermin Ozkiranartli <nermin@google.com>
Owen Carlsen <ocarlsen@gmail.com>
Paneendra Ba <paneendra@gmail.com>
Paul Menage <menage@gmail.com>
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [2007] Neal Norwitz

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

# This file contains a list of people who've made non-trivial contribution to the Google C++ Testing Framework project. People who commit code to the project are encouraged to add their names here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@google.com>
Ji Sigursson <joi@gmail.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@google.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@google.com>
Patrick Riley <spfr@google.com>
Peter Kaminski <piotrk@google.com>
Preston Jackson <preston.a.jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@google.com>
Russ Rufer <russ@pentad.com>
Sean Mcafee <eefacm@gmail.com>
Sigurur sgeirsson <sigg@google.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@google.com>
Zhanyong Wan <wan@google.com>

1.1233 libxcb 1.11.1 1ubuntu1

1.1233.1 Available under license:
Copyright (C) 2001-2006 Bart Massey, Jamey Sharp, and Josh Triplett.
All Rights Reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or their institutions shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the authors.

1.1234 kerby-config 1.0.0
1.1234.1 Available under license :

Kerby Config
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licenser" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licenser for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licenser or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1235 shelljs 0.3.0 0.3.0
1.1235.1 Available under license:

Copyright (c) 2012, Artur Adib <aadib@mozilla.com>
All rights reserved.

You may use this project under the terms of the New BSD license as follows:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Artur Adib nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ARTUR ADIB BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1236 has-own-deep 1.1.0 1.1.0
1.1236.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1237 batik-rasterizer-ext 1.13

1.1237.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole,
an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1238 uri-js 4.4.1 4.4.1
1.1238.1 Available under license:
Copyright 2011 Gary Court. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY GARY COURT "AS IS" AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL GARY COURT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
special, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be
interpreted as representing official policies, either expressed or implied, of Gary Court.

1.1239 diff2html 2.5.0 2.5.0
1.1239.1 Available under license:
Copyright 2014-2016 Rodrigo Fernandes https://rtfpessoa.github.io/

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1240 ignite-core 2.11.0
1.1240.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptation of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==============================================================================
Apache Ignite Subcomponents:

The Apache Ignite project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

==============================================================================
For SnapTree:

This product bundles SnapTree, which is available under a
"3-clause BSD" license. For details, see https://github.com/nbronson/snaptree/blob/master/LICENSE.

==============================================================================
For JSR 166 classes in "org.jsr166" package
==============================================================================
This product bundles JSR-166 classes which are donated to public domain.
For details, see CC0 1.0 Universal (1.0), Public Domain Dedication,
http://creativecommons.org/publicdomain/zero/1.0/

==============================================================================
For books used for tests in "org.apache.ignite.internal.processors.hadoop.books"
==============================================================================
This code bundles book text files used for testing purposes which contain
the following header:

This eBook is for the use of anyone anywhere at no cost and with
almost no restrictions whatsoever. You may copy it, give it away or
re-use it under the terms of the Project Gutenberg License included
with this eBook or online at www.gutenberg.org

// List of ignite-core module's dependencies provided as a part of this distribution.

==============================================================================
For JSR107 API and SPI (https://github.com/jsr107/jsr107spec) javax.cache:cache-api:jar:1.0.0
==============================================================================
This product bundles JSR107 API and SPI which is available under the following:
JSR-000107 JCACHE 2.9 Public Review - Updated Specification License. For details, see

==============================================================================
For ignite-shmem (http://www.gridgain.com) org.gridgain:ignite-shmem:jar:1.0.0
==============================================================================
This product bundles ignite-shmem which is available under the following:
The Apache Software License, Version 2.0. For details, see http://www.apache.org/licenses/LICENSE-2.0.txt.

==============================================================================
==============================================================================
This product bundles JetBrains Java Annotations which is available under the following:
The Apache Software License, Version 2.0. For details, see http://www.apache.org/license/LICENSE-2.0.txt.

Apache Ignite
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
This software includes code from IntelliJ IDEA Community Edition
Copyright (C) JetBrains s.r.o.
https://www.jetbrains.com/idea/
Licensed under Apache License, Version 2.0.
http://search.maven.org/#artifactdetails%7Corg.jetbrains%7Cannotations%7C13.0%7Cjar

1.1241 convert-source-map 1.7.0

1.1241.1 Available under license:
Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1242 batik 1.13

1.1242.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1243 is-callable 1.2.2
1.1243.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1244 emojis-list 2.1.0 2.1.0
1.1244.1 Available under license:
The MIT License (MIT)

Copyright 2015 Kiko Beats

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.1245 uri-js 4.2.2 4.2.2

1.1245.1 Available under license:

<OWNER> = Regents of the University of California
<ORGANIZATION> = University of California, Berkeley
<YEAR> = 1998

In the original BSD license, both occurrences of the phrase "COPYRIGHT HOLDERS AND CONTRIBUTORS" in
the disclaimer read "REGENTS AND CONTRIBUTORS".

Here is the license template:

Copyright (c) <YEAR>, <OWNER>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1246 buffer-shims 1.0.0 1.0.0

1.1246.1 Available under license:

# Copyright (c) 2016 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.**

1.1247 activation 1.1
1.1247.1 Available under license :

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'The contents of this file are subject to the terms of the Common Development and Distribution License (the "License"). You may not use this file except Copyright 2001-2005 Sun Microsystems, Inc. All Rights Reserved.'

Found license 'Common Development and Distribution License (CDDL) 1.1' in '* The contents of this file are subject to the terms * of the Common Development and Distribution License * (the "License"). You may not use this file except * Copyright 2002-2005 Sun Microsystems, Inc. All Rights Reserved.'

Found license 'Common Development and Distribution License (CDDL) 1.1' in '* The contents of this file are subject to the terms * of the Common Development and Distribution License * (the "License"). You may not use this file except * Copyright 1997-2005 Sun Microsystems, Inc. All Rights Reserved.'

1.1248 activation-api 1.2.0
1.1248.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including
1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling
of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be
given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the
termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR
LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to
this License and to the absence of any warranty; and give any other
recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of
it, thus forming a work based on the Program, and copy and distribute
such modifications or work under the terms of Section 1 above, provided
that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
      stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
      whole or in part contains or is derived from the Program or any part
      thereof, to be licensed as a whole at no charge to all third parties
      under the terms of this License.

   c) If the modified program normally reads commands interactively
      when run, you must cause it, when started running for such
      interactive use in the most ordinary way, to print or display an
      announcement including an appropriate copyright notice and a notice
      that there is no warranty (or else, saying that you provide a
      warranty) and that users may redistribute the program under these
      conditions, and telling the user how to view a copy of this License.
      (Exception: if the Program itself is interactive but does not
      normally print such an announcement, your work based on the Program
      is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program, and
can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on
the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a
storage or distribution medium does not bring the other work under the
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the 
original copyright holder who places the Program under this License may 
add an explicit geographical distribution limitation excluding those 
countries, so that distribution is permitted only in or among countries 
not thus excluded. In such case, this License incorporates the 
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new 
versions of the General Public License from time to time. Such new 
versions will be similar in spirit to the present version, but may 
 differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program 
specifies a version number of this License which applies to it and "any 
later version", you have the option of following the terms and 
conditions either of that version or of any later version published by 
the Free Software Foundation. If the Program does not specify a version 
number of this License, you may choose any version ever published by the 
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free 
programs whose distribution conditions are different, write to the 
author to ask for permission. For software which is copyrighted by the 
Free Software Foundation, write to the Free Software Foundation; we 
sometimes make exceptions for this. Our decision will be guided by the 
two goals of preserving the free status of all derivatives of our free 
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO 
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. 
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR 
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, 
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED 
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE 
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH 
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL 
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN 
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY 
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR 
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL 
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM 
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED 
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF 
The PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingling code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to
further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.1249 babel-plugin-transform-strict-mode
6.24.1

1.1249.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1250 gson 2.8.5

1.1250.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Gson authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/stream/JsonToken.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/stream/JsonScope.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/stream/MalformedJsonException.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/stream/JsonReader.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/stream/JsonWriter.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2010 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonSyntaxException.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/Streams.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2008 Google Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/GsonBuilder.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/annotations/Since.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonSerializationContext.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonElement.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/FieldNamingStrategy.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonParseException.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonObject.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/annotations/SerializedName.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonPrimitive.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/ObjectConstructor.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonArray.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/ExclusionStrategy.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonDeserializationContext.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/DefaultDateTypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/InstanceCreator.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonIOException.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/DefaultDateTypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/gson/DefaultDateTypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/serializes/Expose.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/serializes/Until.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/Excluder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Android Open Source Project
 * Copyright (C) 2012 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");  
 * you may not use this file except in compliance with the License. 
 * You may obtain a copy of the License at 
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software 
 * distributed under the License is distributed on an "AS IS" BASIS, 
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
 * See the License for the specific language governing permissions and 
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/TypeAdapters.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/JsonReaderInternalAccess.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/TypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/TypeAdapterFactory.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/ObjectTypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/JsonTreeWriter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/ArrayTypeAdapter.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/TypeAdapterRuntimeTypeWrapper.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/ReflectiveTypeAdapterFactory.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/CollectionTypeAdapterFactory.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/JsonTreeReader.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/MapTypeAdapterFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 Google Inc.
 * Copyright (C) 2012 Google Inc.
 *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/LinkedTreeMap.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/LinkedHashTreeMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/bind/JsonAdapterAnnotationTypeAdapterFactory.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/annotations/JsonAdapter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
*/
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):

* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonStreamParser.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/JsonParser.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/FieldAttributes.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/LongSerializationPolicy.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):

* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/$Gson$Types.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2017 The Gson authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/reflect/UnsafeReflectionAccessor.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/reflect/PreJava9ReflectionAccessor.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/JavaVersion.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/reflect/ReflectionAccessor.java
* /opt/cola/permits/1000738091_1646171488.9/0/gson-2-8-5-sources-jar/com/google/gson/internal/PreJava9DateFormatProvider.java

1.1251 split 0.3.3 0.3.3

1.1251.1 Available under license:
Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1252 @babel/plugin-syntax-numeric-separator 7.10.4

1.1252.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1253 javaparser-core 3.18.0

1.1253.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.
If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy’s public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1254 amqplib-tutorials 0.0.1 0.0.1

1.1254.1 Available under license:

amqplib copyright (c) 2013, 2014
Michael Bridgen <mikeb@squaremobius.net>

This package, "amqplib", is licensed under the MIT License. A copy may be found in the file LICENSE-MIT in this directory, or downloaded from http://opensource.org/licenses/MIT.
The MIT License (MIT)

Copyright (c) 2013, 2014 Michael Bridgen <mikeb@squaremobius.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1255 source-map 0.2.0

1.1255.1 Available under license:

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1256 statuses 1.4.0 1.4.0

1.1256.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1257 avro 1.7.7

1.1257.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the
Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1258 cliui 5.0.0
1.1258.1 Available under license:
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES
1.1259 static-extend 0.1.2 0.1.2

1.1259.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1260 loader-utils 1.2.3 1.2.3

1.1260.1 Available under license:

Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1261 regenerator-runtime 0.13.7 0.13.7
1.1261.1 Available under license :
MIT License

Copyright (c) 2014-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1262 batik-constants 1.13
1.1262.1 Available under license :
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1135872551_1613620997.23/0/batik-constants-1-13-sources-1-jar/org/apache/batik/constants/XMLConstants.java

1.1263 lowercase-keys 1.0.1

1.1263.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1264 snapdragon-util 3.0.1 3.0.1

1.1264.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1265 negotiator 0.6.1 0.6.1

1.1265.1 Available under license :

(The MIT License)

Copyright (c) 2012-2014 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter
Copyright (c) 2014-2015 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1266 lodash.defaults 4.2.0 4.2.0

1.1266.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1267 to-regex-range 5.0.1 5.0.1

1.1267.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.1268 node-extend-without-arrays 1.1.6 1.1.6
1.1268.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*! 
 * node.extend 
 * Copyright 2011, John Resig 
 * Dual licensed under the MIT or GPL Version 2 licenses. 
 * http://jquery.org/license 
 * 
 * @fileoverview 
 * Port of jQuery.extend that actually works on node.js 
 */

Found in path(s):
* /opt/cola/permits/1026169152_1615296591.7/0/node-extend-1-1-6-zip/node.extend-1.1.6/lib/extend.js
No license file was found, but licenses were detected in source scan.

# node.extend

A port of jQuery.extend that **actually works** on node.js

[![Build Status][travis-svg]][travis-url] 
[![dependency status][deps-svg]][deps-url] 
[![dev dependency status][dev-deps-svg]][dev-deps-url] 

[![browser support][testling-png]][testling-url]

### Description

None of the existing ones on npm really work therefore I ported it myself.

### Usage

To install this module in your current working directory (which should already contain a package.json), run

```bash
---
npm install node.extend
---
```

You can additionally just list the module in your [package.json](https://npmjs.org/doc/json.html) and run npm
install.

Then, require this package where you need it:

```javascript
var extend = require('node.extend');
```

The syntax for merging two objects is as follows:

```javascript
var destObject = extend({}, sourceObject);
// Where sourceObject is the object whose properties will be copied into another.
// NOTE: In this situation, this is not a deep merge. See below on how to handle a deep merge.
```

For information about how the clone works internally, view source in lib/extend.js or checkout the doc from [jQuery][1]

### A Note About Deep Merge (avoiding pass-by-reference cloning)

In order to force a deep merge, when extending an object, you must pass boolean true as the first argument to extend:

```javascript
var destObject = extend(true, {}, sourceObject);
// Where sourceObject is the object whose properties will be copied into another.
```

See [this article](http://www.jon-carlos.com/2013/is-javascript-call-by-value-or-call-by-reference/) for more information about the need for deep merges in JavaScript.

## Credit

- Jordan Harband [@ljharb][2]

## License

Copyright 2011, John Resig
Dual licensed under the MIT or GPL Version 2 licenses.
http://jquery.org/license

[testling-png]: https://ci.testling.com/dreamerslab/node.extend.png
[testling-url]: https://ci.testling.com/dreamerslab/node.extend
[travis-svg]: https://travis-ci.org/dreamerslab/node.extend.svg
[travis-url]: https://travis-ci.org/dreamerslab/node.extend
1.1269 lodash-some 4.6.0 4.6.0

1.1269.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1270 jsreport-fs-store 2.9.1 2.9.1

1.1270.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".
The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1271 is-fullwidth-code-point 2.0.0

1.1271.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1272 readdirp 3.2.0

1.1272.1 Available under license:
MIT License

Copyright (c) 2012-2019 Thorsten Lorenz, Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1273 jsp 2.1
1.1273.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
Version 1.0

* 

1. Definitions.

   1.1. ?Contributor? means each individual or entity that creates or contributes to the creation of Modifications.

   1.2. ?Contributor Version? means the combination of the Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

   1.3. ?Covered Software? means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

   1.4. ?Executable? means the Covered Software in any form other than Source Code.

   1.5. ?Initial Developer? means the individual or entity that first makes Original Software available under this
License.

   1.6. ?Larger Work? means a work which combines Covered Software or portions thereof with code not
 governed by the terms of this License.

   1.7. ?License? means this document.

   1.8. ?Licensable? means having the right to grant, to the maximum extent possible, whether at the time of the
 initial grant or subsequently acquired, any and all of the rights conveyed herein.

   1.9. ?Modifications? means the Source Code and Executable form of any of the following:

      A. Any file that results from an addition to, deletion from or modification of the contents of a file
containing Original Software or previous Modifications;

      B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. ?Original Software? means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. ?Patent Claims? means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. ?Source Code? means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. ?You? (or ?Your?) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, ?You? includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, ?control? means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You
represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients? rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient?s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN ?AS IS? BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as ?Participant?) alleging that the Participant Software (meaning the Contributor Version where the Participant is a
Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a ?commercial item,? as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of ?commercial computer software? (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and ?commercial computer software documentation? as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction?s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts.
located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing
party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses.
The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly
excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter
shall not apply to this License. You agree that You alone are responsible for compliance with the United States
export administration regulations (and the export control laws and regulation of any other countries) when You use,
distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising,
directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer
and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be
deemed to constitute any admission of liability.

1.274 babel-plugin-syntax-dynamic-import

6.18.0

1.274.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

MIT License

Copyright (c) 2014-2016 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

BSD License

For "regenerator" software

Copyright (c) 2014, Facebook, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1275 formidable 1.2.2 1.2.2

1.1275.1 Available under license:

The MIT License (MIT)

Copyright (c) 2011-present Felix Geisendrfer, and contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1276 protobuf-java 2.5.0
1.1276.1 Available under license:

No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/Utf8.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/MessageLite.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/Service.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/TextFormat.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/LiteralByteString.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/FieldSet.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/BlockingService.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/UnmodifiableLazyStringList.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/WireFormat.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/MessageOrBuilder.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/InvalidProtocolBufferException.java
* /opt/cola/permits/1000966244_1646171545.31/0/protobuf-java-2-5-0-sources-jar/com/google/protobuf/ExtensionRegistry.java
1.1277 checker-qual 2.10.0

1.1277.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1278 hbase-hadoop-compat 2.0.0-alpha4

1.1278.1 Available under license:

Apache HBase - Hadoop Two Compatibility
Copyright 2007-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications; or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1279 glob-parent 5.1.1
1.1279.1 Available under license:
The ISC License

Copyright (c) 2015, 2019 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1280 foreground-child 2.0.0
1.1280.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.1281 commons-text 1.6

1.1281.1 Available under license:

Apache Commons Text
Copyright 2014-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.1282 hive-serde 3.1.2
1.1282.1 Available under license:

Hive Serde
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1283 zero-allocation-hashing 0.9
1.1283.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Higher Frequency Trading http://www.higherfrequencytrading.com
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/Primitives.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/HotSpotPrior7u6StringHash.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/UnsafeAccess.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/package-info.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/ModernHotSpotStringHash.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/CharSequenceAccess.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/MurmurHash_3.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/CityAndFarmHash_1_1.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/ByteBufferAccess.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/UnknownJvmStringHash.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/StringHash.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/LongHashFunction.java
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/Access.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2015 Higher Frequency Trading http://www.higherfrequencytrading.com
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/net/openhft/hashing/XxHash.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 Higher Frequency Trading http://www.higherfrequencytrading.com
~
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~ 
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.

Found in path(s):
* /opt/cola/permits/1258286306_1643052511.5/0/zero-allocation-hashing-0-9-sources-jar/META-INF/maven/net.openhft/zero-allocation-hashing/pom.xml

1.1284 accepts 1.3.3 1.3.3
1.1284.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1285 jsreport-client 1.2.0 1.2.0

1.1285.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1286 wayland 1.12.0-1~ubuntu16.04.3

1.1286.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next
paragraph) shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

---

The above is the version of the MIT "Expat" License used by X.org:

http://cgit.freedesktop.org/xorg/xserver/tree/COPYING

1.1287 commons-lang3 2.4

1.1287.1 Available under license:

Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

BSD License

For "regenerator" software

Copyright (c) 2014, Facebook, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2014-2016 Sebastian McKenzie <sebmck@gmail.com>
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1289 commons-logging 1.1.1
1.1289.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

// ------------------------------------------------------------------
// NOTICE file corresponding to the section 4d of The Apache License,
// Version 2.0, in this case for Commons Logging
// ------------------------------------------------------------------

Commons Logging
Copyright 2001-2007 The Apache Software Foundation

This product includes/uses software(s) developed by 'an unknown organization'
- Unnamed - avalon-framework:avalon-framework:jar:4.1.3
- Unnamed - log4j:log4j:jar:1.2.12
- Unnamed - logkit:logkit:jar:1.0.1

1.1290 hadoop-hdfs 3.2.2
1.1290.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4,
which has the following notices:

The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.
This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
* license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
* http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
* http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
* license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
* http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
* license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
* https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
* license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
* http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
* license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
* license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
* http://archive.apache.org/dist/harmony/
This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE: * license/LICENSE.abstractnodequeue.txt (Public Domain)

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE: * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE: * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE: * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE: * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE: * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE: * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE: * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE: * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE: * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
* license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4j', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * LICENSE/license.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * LICENSE/license.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.
Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:
* ResolverUtil.java
The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls
  Released under the Creative Commons Attribution License
  (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net
  Any republication or derived work distributed in source code form
  must include this copyright and license notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse
The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.
The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUntime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* =========================================================================
  == NOTICE file corresponding to section 4(d) of the Apache License, ==
  == Version 2.0, in this case for the Apache Xerces Java distribution. ==
  =========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.
The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:

* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak
Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1, which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of joda-time:joda-time:2.9.9 which has the following notices:
* =============================================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  ==============================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  ==============================================================
  This product includes software developed by Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of Ehcache 3.3.1. which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of snakeyaml (https://bitbucket.org/asomov/snakeyaml), which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of swagger-annotations (https://github.com/swagger-api/swagger-core), which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of metrics-core 3.2.4 which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work.
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright 
notices and license terms. Your use of the source code for the these 
subcomponents is subject to the terms and conditions of the following 
licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL
 * nor the names of its contributors may be used to endorse or
 * promote products derived from this software without specific prior
 * written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
 * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
 * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 */

For portions of the native implementation of slicing-by-8 CRC calculation 
in src/main/native/src/org/apache/hadoop/util:
Copyright (c) 2008, 2009, 2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++
 *
* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
* 
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html
* 
* Abstract: The main routine
* 
* ...
*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/[lz4.h,lz4.c,lz4hc.h,lz4hc.c],

/*
LZ4 - Fast LZ compression algorithm
*/
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

* /

For hadoop-common-project/hadoop-common/src/main/native/gtest

-----------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

-------------------------------------------------------------------------------
(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h
----------------------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

============
The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following
license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy
(http://code.google.com/p/snappy/), which is available under the following
license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
  * Neither the name of Google Inc. nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
  hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
--------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js
--------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Copyright is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this
file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the
License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
express or implied. See the License for the specific language governing permissions and
limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

-----------------------------------------------

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Open Source Used In DNAC Platform Arica 4746

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

====
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

---------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
- HSQLDB Database 2.3.4

(HSQL License)
“COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group.

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

Common Development and Distribution License (CDDL) Version 1.0
1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or
hereafter acquired, including without limitation, method, process, and apparatus
claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which
modifications are made and (b) associated documentation included in or
with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights
under, and complying with all of the terms of, this License. For legal entities,
You includes any entity which controls, is controlled by, or is under common control
with You. For purposes of this definition, control means (a) the power, direct
or indirect, to cause the direction or management of such entity, whether by
contract or otherwise, or (b) ownership of more than fifty percent (50%) of the
outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance
with Section 3.1 below and subject to third party intellectual property claims,
the Initial Developer hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than
patent or trademark) Licensable by Initial Developer, to use, reproduce, modify,
display, perform, sublicense and distribute the Original Software (or portions
thereof), with or without Modifications, and/or as part of a Larger Work;
and

(b) under Patent Claims infringed by the making, using or selling of
Original Software, to make, have made, use, practice, sell, and offer for sale,
and/or otherwise dispose of the Original Software (or portions
thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are
effective on the date Initial Developer first distributes or otherwise makes the
Original Software available to a third party under the terms of this
License;

(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or (2) for
infringements caused by: (i) the modification of the Original Software, or
(ii) the combination of the Original Software with other software or
devices.

2.2. Contributor Grant. Conditioned upon Your compliance with
Section 3.1 below and subject to third party intellectual property claims, each
Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent
Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must
include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1.
New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use,
distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under
Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this
License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21
--------------------------------------------------------------------------------
(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made
by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial
Developer, to use, reproduce, modify, display, perform, sublicense and
distribute the Original Software (or portions thereof), with or without
Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims
infringed by the making, using or selling of Original Software, to make, have
made, use, practice, sell, and offer for sale, and/or otherwise dispose of the
Original Software (or portions thereof).
(c) The licenses granted in Sections
2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms
of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or (2) for
infringements caused by: (i) the modification of the Original Software, or (ii)
the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and
subject to third party intellectual property claims, each Contributor hereby
grants You a world-wide, royalty-free, non-exclusive license:
(a) under
intellectual property rights (other than patent or trademark) Licensable by
Contributor to use, reproduce, modify, display, perform, sublicense and
distribute the Modifications created by such Contributor (or portions thereof),
either on an unmodified basis, with other Modifications, as Covered Software
and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such
combination), to make, use, sell, offer for sale, have made, and/or otherwise
dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.
3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You
may create a Larger Work by combining Covered Software with other code not
governed by the terms of this License and distribute the Larger Work as a single
product. In such a case, You must make sure the requirements of this License are
fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or
new versions of this License from time to time. Each version will be given a
distinguishing version number. Except as provided in Section 4.3, no one other
than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the
Covered Software available under the terms of the version of the License under
which You originally received the Covered Software. If the Initial Developer
includes a notice in the Original Software prohibiting it from being distributed
or otherwise made available under any subsequent version of the License, You
must distribute and make the Covered Software available under the terms of the
version of the License under which You originally received the Covered Software.
Otherwise, You may also choose to use, distribute or otherwise make the Covered
Software available under the terms of any subsequent version of the License
published by the license steward.

4.3. Modified Versions.
When You are an
Initial Developer and You want to create a new license for Your Original
Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward
(except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS
LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED
SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY
RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF
WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED
SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will
terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:
- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation,
with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by
that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor,
and informs licensees how to obtain it in a reasonable manner on or through a
medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the
Program.

Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore, if
a Contributor includes the Program in a commercial product offering, such
Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify
every other Contributor (“Indemnified Contributor”) against any losses, damages
and costs (collectively “Losses”) arising from claims, lawsuits and other legal
actions brought by a third party against the Indemnified Contributor to the
extent caused by the acts or omissions of such Commercial Contributor in
connection with its distribution of the Program in a commercial product
offering. The obligations in this section do not apply to any claims or Losses
relating to any actual or alleged intellectual property infringement. In order
to qualify, an Indemnified Contributor must: a) promptly notify the Commercial
Contributor in writing of such claim, and b) allow the Commercial Contributor to
control, and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may participate in
any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If that
Commercial Contributor then makes performance claims, or offers warranties
related to Product X, those performance claims and warranties are such
Commercial Contributor’s responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial
Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using and
distributing the Program and assumes all risks associated with its exercise of
rights under this Agreement, including but not limited to the risks and costs
of program errors, compliance with applicable laws, damage to or loss of data,
programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable
law, it shall not affect the validity or enforceability of the remainder of the
terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such
provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware)
infringes such Recipient's patent(s), then such Recipient's rights granted under
Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to
comply with any of the material terms or conditions of this Agreement and does
not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement
and any licenses granted by Recipient relating to the Program shall continue and
survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be
modified in the following manner. The Agreement Steward reserves the right to
publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

--------------------------------------------------------------------------------
(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.
--------------------------------------------------------------------------------
(2-clause BSD)
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the
following license:
"Java Concurrency in Practice" book annotations 1.0
--------------------------------------------------------------------------------
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS
PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR
OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE
BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.
For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium
or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated.
above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective
Work, the Licensor offers to the recipient a license to the Work on the same
terms and conditions as the license granted to You under this License.
Each time You distribute or publicly digitally perform a Derivative Work,
Licensor offers to the recipient a license to the original Work on the same
terms and conditions as the license granted to You under this License.
If any provision of this License is invalid or unenforceable under applicable
law, it shall not affect the validity or enforceability of the remainder of the
terms of this License, and without further action by the parties to this
agreement, such provision shall be reformed to the minimum extent necessary to
make such provision valid and enforceable.
No term or provision of this License shall be deemed waived and no breach
consented to unless such waiver or consent shall be in writing and signed by the
party to be charged with such waiver or consent.
This License constitutes the entire agreement between the parties with respect
to the Work licensed here. There are no understandings, agreements or
representations with respect to the Work not specified here. Licensor shall not
be bound by any additional provisions that may appear in any communication from
You. This License may not be modified without the mutual written agreement of
the Licensor and You.

The binary distribution of this product bundles these dependencies under the
following license:
jamon-runtime 2.4.1
--------------------------------------------------------------------------------
(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation
of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a
Contributor and that particular Contributors Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice
in Exhibit A, the Executable Form of such Source Code Form, and Modifications of
such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.
1.14. You (or Your) means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.
This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.
5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the
possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses
If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

/*--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission
from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the
software itself an acknowledgement equivalent to the following:

"This product includes software developed by the
JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the JDOM Project and was originally
created by Jason Hunter <jhunter_AT_jdom_DOT_org> and
Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information
on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the
following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is
available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN
ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES
REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR
DAMAGES RESULTING FROM ITS USE.
License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers,
musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the
3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish." or a modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:
   i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
   ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
   iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource
Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your
use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the
Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice
Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1291.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1292 type-fest 0.8.1
1.1292.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Creative Commons Legal Code

CC0 1.0 Universal

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS DOCUMENT DOES NOT CREATE AN
Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

   i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
   ii. moral rights retained by the original author(s) and/or performer(s);
   iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
   iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;
   v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
   vi. database rights (such as those arising under Directive 96/9/EC of the
European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive; and
vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.
a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.
b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.
c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.
d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

1.1293 reselect 3.0.1 3.0.1

1.1293.1 Available under license : The MIT License (MIT)

Copyright (c) 2015-2016 Reselect Contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1294 moment-timezone 0.5.31 0.5.31

1.1294.1 Available under license:

The MIT License (MIT)

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1295 drill-storage-splunk 1.19.0

1.1295.1 Available under license:

Drill : Contrib : Storage : Splunk
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1296 json-smart 2.3

1.1296.1 Available under license:

No license file was found, but licenses were detected in source scan.

```java
/*@ 
* Copyright 2011 JSON-SMART authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONObject.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONNavi.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONArray.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONStreamAwareEx.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/writer/FakeMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONAwareEx.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONValue.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/ParseException.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JStrylerObj.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONUtil.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/JSONParserString.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JsonReader.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/JSONParserReader.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/writer/BeansMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONStreamAware.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONValue.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JsonArray.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONAware.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/JSONParser.java
*/
jar/net/minidev/json/parser/JSONParserBase.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/DefaultMapperOrdered.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/parser/JSONParserMemory.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/ArraysMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/CollectionMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/CompressorMapper.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/DefaultMapperCollection.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/JsonReaderI.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/parser/JSONParserStream.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/JSONStyle.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/parser/JSONParserInputStream.java
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-jar/net/minidev/json/parser/JSONParserByteArray.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2014 JSON-SMART authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177952445_1625583906.57/0/json-smart-2-3-sources-4-
jar/net/minidev/json/writer/DefaultMapper.java
1.1297 set-blocking 2.0.0 2.0.0
1.1297.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1298 duplexify 3.7.1 3.7.1
1.1298.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1299 lodash-es 4.17.15 4.17.15

1.1299.1 Available under license:

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

**1.1300 jackson 2.12.3**

1.1300.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1301 aws-java-sdk-kms 1.11.375

1.1301.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
Open Source Used In DNAC Platform Arica 4807

* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions
* and limitations under the License.
* /

Found in path(s):
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/CreateGrantRequest.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/InvalidAliasNameException.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/DecryptResult.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/TagResourceRequestMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ListAliasesRequestMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/GetParametersForImportResultJsonUnmarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ScheduleKeyDeletionRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/RevokeGrantResultJsonUnmarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ExpiredImportTokenException.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/OriginType.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/EnableKeyRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/RetireGrantRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ListKeyPoliciesRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/AliasListEntryJsonUnmarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ListRetirableGrantsResult.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/ImportKeyMaterialRequest.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-jar/com/amazonaws/services/kms/model/CreateAliasResultJsonUnmarshaller.java
jar/com/amazonaws/services/kms/model/DeleteImportedKeyMaterialRequest.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/ListKeysRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/update/AliasResult.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/GrantConstraintsMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/ScheduleKeyDeletionResultJsonUnmarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/AliasListEntry.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/GenerateRandomRequestProtocolMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/AWSKMSClientBuilder.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/AliasListEntryMarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/CreateAliasRequest.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/EnableKeyRotationResult.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/KeyManagerType.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/CancelKeyDeletionResult.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyResultJsonUnmarshaller.java
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
jar/com/amazonaws/services/kms/model/transform/CreateGrantRequestProtocolMarshaller.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/transform/GenerateDataKeyWithoutPlaintextRequestProtocolMarshaller.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/EnableKeyRequest.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/InvalidImportTokenException.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/PutKeyPolicyResult.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/transform/EnableKeyRotationRequestProtocolMarshaller.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/EncryptResult.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/ScheduleKeyDeletionRequest.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/CreateGrantResult.java

* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/transform/GetKeyRotationStatusResultJsonUnmarshaller.java


* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/NotFoundException.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/AWSKMSException.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/AWSKMSAsync.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/transform/CreateKeyResultJsonUnmarshaller.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/ListGrantsResult.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/ScheduleKeyDeletionResult.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/RevokeGrantRequest.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/ListAliasesRequest.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/ListAliasesResult.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/KMSInternalException.java

* /opt/ cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1.jar/com/amazonaws/services/kms/model/RevokeGrantResult.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135892260_1613626809.29/0/aws-java-sdk-kms-1-11-375-sources-1-
  jar/com/amazonaws/auth/policy/actions/KMSActions.java

1.1302 slash 3.0.0 3.0.0

1.1302.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1303 registry-auth-token 4.2.1
1.1303.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Espen Hovlandsdal

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1304 libbsd 0.8.2 1ubuntu0.1
1.1304.1 Available under license:
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Files:
*

Copyright:
Copyright 2004-2006, 2008-2015 Guillem Jover <guillem@hadrons.org>
License: BSD-3-clause

Files:
man/arc4random.3
man/tree.3

Copyright:
Copyright 1997 Niels Provos <provos@physnet.uni-hamburg.de>
All rights reserved.
License: BSD-4-clause-Niels-Provos
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must display the following acknowledgement:
   This product includes software developed by Niels Provos.
4. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files:
man/getprogname.3
Copyright:
Copyright 2001 Christopher G. Demetriou
All rights reserved.
License: BSD-4-clause-Christopher-G-Demetriou
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must display the following acknowledgement:
      This product includes software developed for the
      NetBSD Project. See http://www.netbsd.org/ for
      information about NetBSD.
4. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files:
include/bsd/err.h
include/bsd/stdlib.h
include/bsd/unistd.h
src/bsd_getopt.c
src/err.c
src/fgetln.c
src/progname.c

Copyright:
Copyright 2005, 2008-2012 Guillem Jover <guillem@hadrons.org>
Copyright 2005 Hector Garcia Alvarez
Copyright 2005 Aurelien Jarno
Copyright 2006 Robert Millan
License: BSD-3-clause

Files:
include/bsd/netinet/ip_icmp.h
include/bsd/sys/bitstring.h
include/bsd/sys/queue.h
include/bsd/timeconv.h
include/bsd/vis.h
man/bitstring.3
man/explicit_bzero.3
man/fgetln.3
man/fgetwln.3
man/funopen.3bsd
man/getbsize.3
man/heapsort.3
man/nlist.3
man/queue.3bsd
man/radixsort.3
man/realloccarray.3
man/reallocf.3
man/setmode.3
man/strmode.3
man/strnstr.3
man/unvis.3
man/vis.3
man/wcslepy.3
src/getbsize.c
src/heapsort.c
src/merge.c
src/nlist.c
src/radixsort.c
src/setmode.c
src/strmode.c
src/strnstr.c
src/unvis.c
src/vis.c
Copyright:
The Regents of the University of California. All rights reserved.
Copyright 2001 Mike Barcroft <mike@FreeBSD.org>
Some code is derived from software contributed to Berkeley by
the American National Standards Committee X3, on Information
Processing Systems.
Some code is derived from software contributed to Berkeley by
Peter McIlroy.
Some code is derived from software contributed to Berkeley by
Ronnie Kon at Mindcraft Inc., Kevin Lew and Elmer Yglesias.
Some code is derived from software contributed to Berkeley by
Dave Borman at Cray Research, Inc.
Some code is derived from software contributed to Berkeley by
Paul Vixie.
Some code is derived from software contributed to Berkeley by
Chris Torek.
Copyright UNIX System Laboratories, Inc.
All or some portions of this file are derived from material licensed
to the University of California by American Telephone and Telegraph
Co. or Unix System Laboratories, Inc. and are reproduced herein with
the permission of UNIX System Laboratories, Inc.
License: BSD-3-clause-Regents
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS `AS IS' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files:
include/bsd/libutil.h
Copyright:
Copyright 1996 Peter Wemm <peter@FreeBSD.org>. All rights reserved.
Copyright 2002 Networks Associates Technology, Inc. All rights reserved.
License: BSD-3-clause-Peter-Wemm
Redistribution and use in source and binary forms, with or without modification, is permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files:
man/setproctitle.3
Copyright:
Copyright 1995 Peter Wemm <peter@FreeBSD.org> All rights reserved.
License: BSD-5-clause-Peter-Wemm
Redistribution and use in source and binary forms, with or without modification, is permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice immediately at the beginning of the file, without modification, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. This work was done expressly for inclusion into FreeBSD. Other use is permitted provided this notation is included.
4. Absolutely no warranty of function or purpose is made by the author Peter Wemm.
5. Modifications may be freely made to this file providing the above conditions are met.

Files:
include/bsd/stringlist.h
man/fmtcheck.3
man/humanize_number.3
man/stringlist.3
src/fmtcheck.c
src/humanize_number.c
src/stringlist.c

Copyright:
All rights reserved.

Some code was contributed to The NetBSD Foundation by Allen Briggs.

Some code was contributed to The NetBSD Foundation by Luke Mewburn.


Some code is derived from software contributed to The NetBSD Foundation by Julio M. Merino Vidal, developed as part of Google's Summer of Code 2005 program.

Some code is derived from software contributed to The NetBSD Foundation by Christos Zoulas.

License: BSD-2-clause-NetBSD
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS
`AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Files:
include/bsd/sys/endian.h
man/expand_number.3
man/closefrom.3
man/flopen.3
man/getpeereid.3
man/pidfile.3
src/expand_number.c
src/hash/sha512.h
src/hash/sha512c.c
src/pidfile.c
src/reallocf.c
src/timeconv.c

Copyright:
Copyright 1998, M. Warner Losh <imp@freebsd.org>
All rights reserved.

Copyright 2001 Dima Dorfman.
All rights reserved.

Copyright 2001 FreeBSD Inc.
All rights reserved.

Copyright 2002 Thomas Moestl <tmm@FreeBSD.org>
All rights reserved.

Copyright 2005 Pawel Jakub Dawidek <pjd@FreeBSD.org>
All rights reserved.

Copyright 2005 Colin Percival
All rights reserved.
Files:
src/flopen.c
Copyright:
Copyright 2007 Dag-Erling Codan Smrgrav
All rights reserved.
License: BSD-2-clause-verbatim
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer
   in this position and unchanged.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Files:
include/bsd/sys/tree.h
man/fparseln.3
src/fparseln.c
Copyright:
Copyright 1997 Christos Zoulas.
All rights reserved.

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.
License: BSD-2-clause-author
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files:
include/bsd/readpassphrase.h
man/readpassphrase.3
man/strlcpy.3
man/strtonum.3
src/arc4random.c
src/arc4random_openbsd.h
src/arc4random_uniform.c
src/arc4random_unix.h
src/closefrom.c
src/getentropy_aix.c
src/getentropy_bsd.c
src/getentropy_hpux.c
src/getentropy_hurd.c
src/getentropy_linux.c
src/getentropy_osx.c
src/getentropy_solaris.c
src/readpassphrase.c
src/reallocarray.c
src/strlcarray.c
src/strlcpy.c
src/strlcpy.3
src/strtonum.c
Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the
following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
USE OR OTHER DEALINGS IN THE SOFTWARE.

This code implements the MD5 message-digest algorithm.
The algorithm is due to Ron Rivest. This code was
written by Colin Plumb in 1993, no copyright is claimed.
This code is in the public domain; do with it what you wish.

This code is in the public domain; do with it what you wish.
Copyright:
None
License: Beerware
"THE BEER-WARE LICENSE" (Revision 42):
<phk@login.dkuug.dk> wrote this file. As long as you retain this notice you
can do whatever you want with this stuff. If we meet some day, and you think
this stuff is worth it, you can buy me a beer in return. Poul-Henning Kamp

License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: BSD-2-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1305 c3p0 0.9.5.4
1.1305.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'This library is free software; you can redistribute it and/or modify
1) The GNU Lesser General Public License (LGPL), version 2.1, as published by the Free Software Foundation
2) The Eclipse Public License (EPL), version 1.0 This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. LGPL v2.1: http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html'

Found license 'GNU Lesser General Public License' in 'This library is free software; you can redistribute it and/or modify
1) The GNU Lesser General Public License (LGPL), version 2.1, as published by the Free Software Foundation
2) The Eclipse Public License (EPL), version 1.0 This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. LGPL v2.1: http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html'

1.1306 hive-llap-client 3.1.2
1.1306.1 Available under license:

Hive Llap Client
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1307 mongodb-migrations 1.0.1 1.0.1
1.1307.1 Available under license:

MIT License

Copyright (c) 2018 Droplr, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1308 call-bind 1.0.2

1.1308.1 Available under license:

MIT License

Copyright (c) 2020 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1309 word-wrap 1.2.3

1.1309.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1310 batik-i18n 1.13

1.1310.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

1.1311 xdg-basedir 4.0.0
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1312 which 1.3.1 1.3.1
1.1312.1 Available under license :
   The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1313 my-scope-package-a 0.0.0
1.1313.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!)  The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The JsonCpp library's source code, including accompanying documentation, tests and demonstration applications, are licensed under the following conditions...

The author (Baptiste Lepilleur) explicitly disclaims copyright in all jurisdictions which recognize such a disclaimer. In such jurisdictions, this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of 2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur, and is released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this software may choose to accept it either as 1) Public Domain, 2) under the conditions of the MIT License (see below), or 3) under the terms of dual Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

========================================================================
Copyright (c) 2007-2010 Baptiste Lepilleur

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

========================================================================
(END LICENSE TEXT)

The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the Public Domain "license" you can re-license your copy using whatever license you like.

All files in this directory are provided by the following license if not stated otherwise in the individual file:

====================
This file is provided under a dual BSD/GPLv2 license. When using or redistributing this file, you may do so under either license.

GPL LICENSE SUMMARY

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.

This program is free software; you can redistribute it and/or modify it under the terms of version 2 of the GNU General Public License as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St - Fifth Floor, Boston, MA 02110-1301 USA. The full GNU General Public License is included in this distribution in the file called LICENSE/GPL.

Contact Information:
BSD LICENSE

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This license applies to all parts of V8 that are not externally maintained libraries. The externally maintained libraries used by V8 are:

- PCRE test suite, located in test/mjsunit/third_party/regexp-pcre/regexp-pcre.js. This is based on the test suite from PCRE-7.3, which is copyrighted by the University of Cambridge and Google, Inc. The copyright notice and license are embedded in regexp-pcre.js.

- Layout tests, located in test/mjsunit/third_party/object-keys. These are based on layout tests from webkit.org which are copyrighted by Apple Computer, Inc. and released under a 3-clause BSD license.

- Strongtalk assembler, the basis of the files assembler-arm-inl.h, assembler-arm.cc, assembler-arm.h, assembler-ia32-inl.h,
assembler-ia32.cc, assembler-ia32.h, assembler-x64-inl.h,
assembler-x64.cc, assembler-x64.h, assembler-mips-inl.h,
assembler-mips.cc, assembler-mips.h, assembler.cc and assembler.h.
This code is copyrighted by Sun Microsystems Inc. and released
under a 3-clause BSD license.

- Valgrind client API header, located at src/third_party/valgrind/valgrind.h
This is released under the BSD license.

- The Wasm C/C++ API headers, located at third_party/wasm-api/wasm.{h,hh}
This is released under the Apache license. The API's upstream prototype
implementation also formed the basis of V8's implementation in
src/wasm/c-api.cc.

These libraries have their own licenses; we recommend you read them,
as their terms may differ from the terms below.

Further license information can be found in LICENSE files located in
sub-directories.

Copyright 2014, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 1994-2006 Sun Microsystems Inc.
All Rights Reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Sun Microsystems or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Tests included here are based on build output generated by the six-speed benchmark suite.

Copyright (c) 2015 Kevin Decker

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
CC0 1.0 Universal

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;

ii. moral rights retained by the original author(s) and/or performer(s);

iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;

iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;

v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
vi. database rights (such as those arising under Directive 96/9/EC of the
protection of databases, and under any national implementation thereof,
including any amended or successor version of such directive); and

vii. other similar, equivalent or corresponding rights throughout the world
based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of,
applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and
unconditionally waives, abandons, and surrenders all of Affirmer's Copyright
and Related Rights and associated claims and causes of action, whether now
known or unknown (including existing as well as future claims and causes of
action), in the Work (i) in all territories worldwide, (ii) for the maximum
duration provided by applicable law or treaty (including future time
extensions), (iii) in any current or future medium and for any number of
copies, and (iv) for any purpose whatsoever, including without limitation
commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes
the Waiver for the benefit of each member of the public at large and to the
detriment of Affirmer's heirs and successors, fully intending that such Waiver
shall not be subject to revocation, rescission, cancellation, termination, or
any other legal or equitable action to disrupt the quiet enjoyment of the Work
by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be
judged legally invalid or ineffective under applicable law, then the Waiver
shall be preserved to the maximum extent permitted taking into account
Affirmer's express Statement of Purpose. In addition, to the extent the Waiver
is so judged Affirmer hereby grants to each affected person a royalty-free,
non transferable, non sublicensable, non exclusive, irrevocable and
unconditional license to exercise Affirmer's Copyright and Related Rights in
the Work (i) in all territories worldwide, (ii) for the maximum duration
provided by applicable law or treaty (including future time extensions), (iii)
in any current or future medium and for any number of copies, and (iv) for any
purpose whatsoever, including without limitation commercial, advertising or
promotional purposes (the "License"). The License shall be deemed effective as
of the date CC0 was applied by Affirmer to the Work. Should any part of the
License for any reason be judged legally invalid or ineffective under
applicable law, such partial invalidity or ineffectiveness shall not
validate the remainder of the License, and in such case Affirmer hereby
affirms that he or she will not (i) exercise any of his or her remaining
Copyright and Related Rights in the Work or (ii) assert any associated claims
and causes of action with respect to the Work, in either case contrary to
Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

a. No trademark or patent rights held by Affirmer are waived, abandoned,
surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

For more information, please see
<http://creativecommons.org/publicdomain/zero/1.0/>

Notice that the following BSD-style license applies to this one file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

This file is part of Valgrind, a dynamic binary instrumentation framework.

Copyright (C) 2000-2010 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----------------------------------------------------------------
Notice that the above BSD-style license applies to this one file (valgrind.h) only. The entire rest of Valgrind is licensed under the terms of the GNU General Public License, version 2. See the COPYING file in the source distribution for details.

----------------------------------------------------------------
// Copyright 2016 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 7 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions.

THE BASIC LIBRARY FUNCTIONS
-------------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2007 University of Cambridge
All rights reserved.

THE C++ WRAPPER FUNCTIONS
-------------------------------

Contributed by: Google Inc.

Copyright (c) 2007, Google Inc.
All rights reserved.

THE "BSD" LICENCE
-------------------

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End

Copyright (c) 2006 Apple Computer, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder(s) nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2010 Jonathan Hartley
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors
  may be used to endorse or promote products derived from this software without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

A. HISTORY OF THE SOFTWARE
==================================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.
In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations, which became Zope Corporation. In 2001, the Python Software Foundation (PSF, see https://www.python.org/psf/) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived from</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td></td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
</tr>
<tr>
<td>1.3 thru 1.5.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2 and above</td>
<td>2.1.1</td>
<td>2001-now</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.


3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.
BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee
agrees to be bound by the terms and conditions of this License Agreement.

---

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 (“CNRI”), and the Individual or Organization (“Licensee”) accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013”.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.
6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2
--------------------------------------------------
Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
INTERUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

W3C 3-clause BSD License

http://www.w3.org/Consortium/Legal/2008/03-bsd-license.html

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of works must retain the original copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the original copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the W3C nor the names of its contributors may be used to endorse or promote products derived from this work without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2006-2011, the V8 project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1314 trim-right 1.0.1

1.1314.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1315 ajv 6.12.6 6.12.6

1.1315.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1316 core-util-is 1.0.2 1.0.2

1.1316.1 Available under license :

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1317 datanucleus-core 4.1.17

1.1317.1 Available under license :

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2008-2013 DataNucleus

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

ObjectWeb ASM 4.1 is repackaged in this datanucleus-core jar under org.datanucleus.asm. The files in that package are licensed under the following terms

====================================================================
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

inance

test

This product includes software developed by many individuals,
including the following:

Erik Bengtson
Andy Jefferson

This product has included contributions from some individuals,
including the following:

Joerg von Frantzius
This product also includes software developed by the TJDO project (http://tjdo.sourceforge.net/).

1.1318 negotiator 0.6.2 0.6.2

1.1318.1 Available under license:
(The MIT License)

Copyright (c) 2012-2014 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1319 is-stream 2.0.0

1.1319.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1320 glob-parent 5.1.1 5.1.1

1.1320.1 Available under license:
The ISC License

Copyright (c) 2015, 2019 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1321 base64id 2.0.0 2.0.0
1.1321.1 Available under license:
(The MIT License)

Copyright (c) 2012-2016 Kristian Faeldt <faeldt_kristian@cyberagent.co.jp>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1322 ipaddr.js 1.9.1 1.9.1

1.1322.1 Available under license:

Copyright (C) 2011-2017 whitequark <whitequark@whitequark.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1323 readdirp 3.3.0 3.3.0

1.1323.1 Available under license:
MIT License

Copyright (c) 2012-2019 Thorsten Lorenz, Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1324 electron-to-chromium 1.3.904

1.1324.1 Available under license:
Copyright 2018 Kilian Valkhof

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1325 lodash-pick 4.4.0 4.4.0
1.1325.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.
1.1326 elfutils 0.165 3ubuntu1.2

1.1326.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
to your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.
A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to
control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such
When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your
A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).
The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or
authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have
received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.
A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.
A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:
a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

  0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

  1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1327 jetty 9.4.34.v20201102

1.1327.1 Available under license :

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.

No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and 
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

====================================================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
====================================================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message
The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
Mortbay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1328 onetime 1.1.0
1.1328.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1329 p-reduce 2.1.0 2.1.0
1.1329.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1330 is-fullwidth-code-point 1.0.0 1.0.0
1.1330.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1331 mixin-deep 1.3.2 1.3.2

1.1331.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1332 diff 3.5.0 3.5.0

1.1332.1 Available under license:

Software License Agreement (BSD License)

Copyright (c) 2009-2015, Kevin Decker <kpdecker@gmail.com>

All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Kevin Decker nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1333 mime-db 1.44.0 1.44.0
The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1334 babel-plugin-syntax-async-functions

6.13.0 6.13.0

1.1334.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1335 unpipe 1.0.0 1.0.0

1.1335.1 Available under license:
(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1336 readable-stream 3.6.0 3.6.0

1.1336.1 Available under license:
Node.js is licensed for use as follows:

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

==

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

==

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

==

1.1337 moment 2.22.2 2.22.2

1.1337.1 Available under license:

Copyright (c) JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1338 guava 30.1-jre
1.1338.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Ascii.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Strings.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/GwtTransient.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/AtomicLongMap.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2016 The Guava Authors
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/MoreCollectors.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ImmutableValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ElementOrder.java
Open Source Used In DNAC Platform Arica 4919

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/Comparators.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/GraphBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/UndirectedMultiNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/EndpointPair.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/DirectedNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/ImmutableMultisetGwtSerializationDependencies.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/StandardMutableValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/RangeGwtSerializationDependencies.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of {@code Doubles}. Intended to be empty for regular
 * version.
 */
Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/DoublesMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/**
 * Holder for web specializations of methods of [Ints]. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/IntsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSortedAsList.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/CollectSpliterators.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableBiMapFauxverideShim.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractMultiset.java
Open Source Used In DNAC Platform Arica 4923

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingMapEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Sets.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Multimaps.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/BitMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Multisets.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingConcurrentMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ReverseNaturalOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/NullsFirstOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SingletonImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/EnumHashBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Multiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Ordering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TreeMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/LinkedListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractSortedSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ConcurrentHashMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/HashMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableList.java
Open Source Used In DNAC Platform Arica  4924

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/NullsLastOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Multimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Iterations.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Interner.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Synchronized.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ComparatorOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompoundOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractMapBasedMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingSortedMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/MutableClassToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractMapBasedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/UsingToStringOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Iterables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TreeMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/NaturalOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Lists.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableSet.java
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/Resources.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Function.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Throwables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/Primitives.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/FinalizableWeakReference.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ExecutionList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/Files.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AbstractFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Preconditions.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/Objects.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/ByteStreams.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/ForwardingLock.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/AbstractHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/ForwardingCondition.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/Java8Compatibility.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/OverflowAvoidingLockSupport.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/ToDoubleRounnder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/BigDecimalMath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/Java8Compatibility.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may
 * obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/Streams.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* Copyright (C) 2015 The Guava Authors

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Internal.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2015 The Guava Authors

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/LittleEndianByteArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ReaderInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ConsumingQueueIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AggregateFutureState.java

Open Source Used In DNAC Platform Artica 4930
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/ImmutableSupplier.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ExecutionSequencer.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/IgnoreJRERequirement.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/*
 * This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
 * and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 *
 * As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
 * java.util.HashMap class.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/WrappingExecutorService.java
Open Source Used In DNAC Platform Arica 4933

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/Optional.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/Ticker.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/RegularImmutableMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/Uninterruptibles.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/net/HostAndPort.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/MathPreconditions.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/net/HttpHeaders.java

---

Open Source Used In DNAC Platform Arica  4934
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/BigIntegerMath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/AbstractCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/HashingOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/EmptyContiguousSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/DoubleMath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/Funnels.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/CacheLoader.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/Funnel.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/ForwardingCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/Queues.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/AbstractSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache>LoadingCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/primitives/ParseRequest.java

---

Open Source Used In DNAC Platform Arica 4935
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RegularImmutableSortedMultiset.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSortedMultisetFauxverideShim.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/GeneralRange.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingSortedMultiset.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SortedIterable.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RangeSet.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractRangeSet.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/CountRangeSet.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/Count.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SortedIterables.java
 * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSortedMultiset.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Returns an array containing all of the elements in the specified collection. This method
* returns the elements in the order they are returned by the collection's iterator. The returned
* array is "safe" in that no references to it are maintained by the collection. The caller is
* thus free to modify the returned array.
* 
* <p>This method assumes that the collection size doesn't change while the method is running.
* 
* <p>TODo(kevinb): support concurrently modified collections?
* 
* @param c the collection for which to return an array of elements
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ObjectArrays.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author hereby disclaims copyright to this source code.

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/SmallCharMatcher.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/SubscriberRegistry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/Dispatcher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/MoreObjects.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ListenerCallQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/Quantiles.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/Subscriber.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ServiceManagerBridge.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Java8Usage.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2006 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/FluentFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Futures.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/VisibleForTesting.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Runnables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Verify.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/SubscriberExceptionContext.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/CharSequenceReader.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Utf8.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/IdkBackedImmutableMap.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/IdkBackedImmutableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AllEqualOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TransformedListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredEntrySetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedMultisetBridge.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactLinkedHashSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactLinkedHashMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactHashMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors.
*/

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableEnumMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingImmutableList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingBlockingDeque.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingDeque.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TransformedIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingImmutableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TreeTraverser.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactHashSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/DescendingMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredEntryMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RangeMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingNavigableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TreeRangeMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/DescendingImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableAsList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/EvictingQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingNavigableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractNavigableMap.java

Open Source Used In DNAC Platform Arica  4943

* Copyright (C) 2011 The Guava Authors.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/\n* Copyright (C) 2016 The Guava Authors
*Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
/**
 * Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/\n* Copyright (C) 2007 The Guava Authors
*Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
/**
 * Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/\n* Copyright (C) 2007 The Guava Authors
*Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
*
Open Source Used In DNAC Platform Arica 4945

* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
*/
*/
* This following method is a modified version of one found in
* http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
* which contained the following notice:
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
* the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ImmutableMapEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/MultimapBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/MoreFiles.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/Tables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/Serialization.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/StandardTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/PeekingIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RegularImmutableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SingletonImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RegularImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableMapValues.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ImmutableList.java
/*
   * Copyright (C) 2020 The Guava Authors
   *
   * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
   * in compliance with the License. You may obtain a copy of the License at
   *
   *      http://www.apache.org/licenses/LICENSE-2.0
   *
   * Unless required by applicable law or agreed to in writing, software distributed under the License
   * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
   * express
   * or implied. See the License for the specific language governing permissions and limitations under
   * the License.
   */
/**
 * Holder for web specializations of methods of [code Shorts]. Intended to be empty for regular
 * version.
 */
/*

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/ShortsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

/*

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableRangeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/Closer.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/Stats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/html/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/math/PairedStatsAccumulator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/Element.java
*/
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/SipHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/common/base/StandardSystemProperty.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/common/math/PairedStats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ListeningScheduledFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/FileWriteMode.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
/* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/IncidentEdgeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collect/CompactHashing.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collect/Hashing.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2005 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/reflect/Reflection.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/**
 * Outer class that exists solely to let us write [ @code Partially.GwtIncompatible ] instead of plain
 * [ @code GwtIncompatible ]. This is more accurate for [ @link Futures#catching ], which is available
 * under GWT but with a slightly different signature.
 *
 * <p>We can't use [ @code PartiallyGwtIncompatible ] because then the GWT compiler wouldn't recognize
 * it as a [ @code GwtIncompatible ] annotation. And for [ @code Futures.catching ], we need the GWT
 * compiler to autostrip the normal server method in order to expose the special, inherited GWT
 * version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/Partially.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/TopKSelector.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/Network.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/RecursiveDeleteOption.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AtomicDoubleArray.java

No license file was found, but licenses were detected in source scan.

/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AtomicDoubleArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/TreeJoiner.java

---

Open Source Used In DNAC Platform Arica 4953
jar/com/google/common/hash/LongAdder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/Striped64.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/FileBackedOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Shorts.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Doubles.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Booleans.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Chars.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/net/PercentEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/base/CharMatcher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Floats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/base/Joiner.java
jar/com/google/common/util/concurrent/ListenableFutureTask.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/UnicodeEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Bytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/MultiReader.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Stopwatch.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Converter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Longs.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Ints.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/internal/Finalizer.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/InetAddresses.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/Escaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/TrieParser.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/MinMaxPriorityQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractSequentialIterator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not
 * use this file except in compliance with the License. You may obtain a copy of
 * the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedMultisets.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 *
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collections/DenseImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/annotations/GwtIncompatible.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collections/SparseImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collections/MapMakerInternalMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collections/MapMaker.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collections/Cut.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingFluentFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RegularImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Splitter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/Escapers.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2009 The Guava Authors
* * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* * http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * Not supported. <b>You are attempting to create a map that may contain a non-{@code Comparable}
 * key.</b> Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy
 * version.
 *
 * @throws UnsupportedOperationException always
 * @deprecated <b>Pass a key of type {@code Comparable} to use {@link
 *     ImmutableSortedMap#of(Comparable, Object)}.</b>
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collect/ImmutableSortedMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of {@code Floats}. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/FloatsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2009 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ComputationException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/DiscreteDomain.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableAsList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ArrayTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TableCollectors.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableEnumSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SingletonImmutableList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SingletonImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ComparisonChain.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableList.java
1.1339 request 2.87.0 2.87.0

1.1339.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/EmptyImmutableSetMultimap.java
or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the
Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright
owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written
communication sent to the Licensor or its representatives, including but not limited to communication on electronic
mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously
marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been
received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section)
patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent
infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date
such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any
medium, with or without modifications, and in Source or Object form, provided that You meet the following
conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License. You may add Your own copyright statement to Your modifications and may provide
additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1340 janino 3.0.11

1.1341 ansi-styles 4.2.1

1.1341.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1342 find-up 3.0.0
1.1342.1 Available under license :
   MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1343 zookeeper 3.5.7
1.1343.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1344.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1345 jackson-databind 2.12.1

1.1345.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1346 cliui 7.0.4 7.0.4
1.1346.1 Available under license:

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1347 eslint-plugin-import 2.25.3
1.1347.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Ben Mosher

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1348 jcip-annotation 1.0 1
1.1348.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 Stephen Connolly.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-jar/net/jcip/annotations/NotThreadSafe.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-jar/net/jcip/annotations/ThreadSafe.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-jar/net/jcip/annotations/GuardedBy.java
* /opt/ws_local/PERMITS_SQL/1013822089_1591370825.53/0/jcip-annotations-1-0-1-sources-jar/net/jcip/annotations/Immutable.java

1.1349 mimic-fn 1.2.0 1.2.0
1.1349.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1350 is-number-object 1.0.4

1.1350.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1351 q 0.9.7 0.9.7

1.1351.1 Available under license :

Copyright 20092012 Kristopher Michael Kowal. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1352 jakarta-validation-api 2.0.2

1.1352.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Jakarta Bean Validation API
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */

Found in path(s):
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/Validator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/CascadableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/ConstraintTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraints/FutureOrPresent.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraints/Null.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/constraints/ConstraintValidatorContext.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/ExecutableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/groups/Default.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/MessageInterpolator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/package-info.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/javax/validation/metadata/package-info.java
Open Source Used In DNAC Platform Arica 4978
No license file was found, but licenses were detected in source scan.

~ Jakarta Bean Validation API
~
~ License: Apache License, Version 2.0
~ See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>
<![[CDATA[
Comments to: <a href="mailto:bean-validation-dev@eclipse.org">bean-validation-dev@eclipse.org</a>.<br>
Copyright &© 2019 Eclipse Foundation.<br>
Use is subject to <a href="" target="_top">EFSL</a>; this spec is based on material that is licensed under the Apache License, version 2.0.]]>

Found in path(s):
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/META-INF/maven/jakarta.validation/jakarta.validation-api/pom.xml
1.1353 graphite 1.3.10-0ubuntu0.16.04.1
1.1353.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations
below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it
becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must
be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference
Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

/* GRAPHITE2 LICENSING */

Copyright 2010, SIL International
All rights reserved.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should also have received a copy of the GNU Lesser General Public
License along with this library in the file named "LICENSE".
If not, write to the Free Software Foundation, 51 Franklin Street,
Suite 500, Boston, MA 02110-1335, USA or visit their web page on the

Alternatively, you may use this library under the terms of the Mozilla
Public License (http://mozilla.org/MPL) or under the GNU General Public
License, as published by the Free Software Foundation; either version
2 of the license or (at your option) any later version.

Format: http://dep.debian.net/deps/dep5
Upstream-Name: graphite2
Source: http://sf.net/projects/silgraphite

Files: *
Copyright: (C) 2010-2011, SIL International
License: LGPL-2+
   This library is free software; you can redistribute it and/or
   modify it under the terms of the GNU Library General Public
   License as published by the Free Software Foundation; either
   version 2 of the License, or (at your option) any later version.

   This library is distributed in the hope that it will be useful,
   but WITHOUT ANY WARRANTY; without even the implied warranty of
   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
   Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library
General Public License in /usr/share/common-licenses/LGPL-2. If you
downloaded the source you will find it in the file LICENSE; if not,
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Files: contrib/perl/*
Copyright: Copyright (C) 2011 Simon Cozens <simon@cpan.org>
License: GPL1+, Artistic
   This program is free software; you can redistribute it and/or modify
   it under the terms of either:

   a) the GNU General Public License as published by the Free Software
      Foundation; either version 1, or (at your option) any later
      version, or

   b) the "Artistic License" which comes with Perl.

On Debian GNU/Linux systems, the complete text of the GNU General
Public License can be found in ‘/usr/share/common-licenses/GPL’ and
the Artistic Licence in ‘/usr/share/common-licenses/Artistic’.
Files: contrib/perl/ppport.h
Copyright: 2004-2010, Marcus Holland-Moritz <mhx-cpan@gmx.net>
2001, Paul Marquess <pmqs@cpan.org> (Version 2.x)
1999, Kenneth Albanowski <kjahds@kjahds.com> (Version 1.x)
License: Artistic or GPL-1+

Files: debian-src/*
Copyright: 2010 Martin Hosken <martin_hosken@sil.org>
License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library General Public License in /usr/share/common-licenses/LGPL-2. If you downloaded the source you will find it in the file LICENSE; if not, along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Files: debian/*/*
Copyright: 2010 Martin Hosken <martin_hosken@sil.org>
2011 Rene Engelhard <renee@debian.org>
License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library General Public License in /usr/share/common-licenses/LGPL-2. If you downloaded the source you will find it in the file LICENSE; if not, along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Files: tests/comparerenderer/icule/*
Copyright: Copyright (c) 1995-2006 International Business Machines Corporation and others
License: other
Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

Files: gr2fonttest/gr2FontTest.cpp
Copyright: Copyright (C) 2005 www.thanlwinsoft.org
License: LGPL | other

Files: tests/comparerenderer/GrUtfTextSrc.*
Copyright: Copyright (C) 2004 Keith Stribley
License: MPL-1.1 | GPL-2 | LGPL-2.1

Files: tests/fonts/*
License: other

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide
development of collaborative font projects, to support the font creation
efforts of academic and linguistic communities, and to provide a free and
open framework in which fonts may be shared and improved in partnership
with others.

The OFL allows the licensed fonts to be used, studied, modified and
redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.
4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL,. INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.
Format: http://dep.debian.net/deps/dep5
Upstream-Name: graphite2
Source: http://sf.net/projects/silgraphite

Files: *
Copyright: (C) 2010-2011, SIL International
License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library General Public License in /usr/share/common-licenses/LGPL-2. If you downloaded the source you will find it in the file LICENSE; if not, along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Files: contrib/perl/*
Copyright: Copyright (C) 2011 Simon Cozens <simon@cpan.org>
License: GPL1+ | Artistic

This program is free software; you can redistribute it and/or modify
it under the terms of either:

a) the GNU General Public License as published by the Free Software
Foundation; either version 1, or (at your option) any later
version, or

b) the "Artistic License" which comes with Perl.

On Debian GNU/Linux systems, the complete text of the GNU General
Public License can be found in `/usr/share/common-licenses/GPL' and
the Artistic Licence in `/usr/share/common-licenses/Artistic'.

Files: contrib/perl/ppport.h
Copyright: 2004-2010, Marcus Holland-Moritz <mhx-cpan@gmx.net>
2001, Paul Marquess <pmqs@cpan.org> (Version 2.x)
1999, Kenneth Albanowski <kjahds@kjahds.com> (Version 1.x)
License: Artistic or GPL-1+

Files: debian-src/*
Copyright: 2010 Martin Hosken <martin_hosken@sil.org>
License: LGPL-2+

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library
General Public License in /usr/share/common-licenses/LGPL-2. If you
downloaded the source you will find it in the file LICENSE; if not,
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Files: debian*/*
Copyright: 2010 Martin Hosken <martin_hosken@sil.org>
2011 Rene Engelhard <rene@debian.org>
License: LGPL-2+

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

On Debian systems you should have a local copy of the GNU Library General Public License in /usr/share/common-licenses/LGPL-2. If you downloaded the source you will find it in the file LICENSE; if not, along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Files: tests/comparerenderer/icule/*
Copyright: Copyright (c) 1995-2006 International Business Machines Corporation and others
License: other

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Files: gr2fonttest/gr2FontTest.cpp
Copyright: Copyright (C) 2005 www.thanlwinsoft.org
License: LGPL | other

Files: tests/comparerenderer/GrUtfTextSrc.*
Copyright: Copyright (C) 2004 Keith Stribley
License: MPL-1.1 | GPL-2 | LGPL-2.1

Files: tests/fonts/*
License: other

---------------------------------

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007
---------------------------------

  .

  PREAMBLE
  The goals of the Open Font License (OFL) are to stimulate worldwide
development of collaborative font projects, to support the font creation
efforts of academic and linguistic communities, and to provide a free and
open framework in which fonts may be shared and improved in partnership
with others.

  .

  The OFL allows the licensed fonts to be used, studied, modified and
redistributed freely as long as they are not sold by themselves. The
fonts, including any derivative works, can be bundled, embedded,
redistributed and/or sold with any software provided that any reserved
names are not used by derivative works. The fonts and derivatives,
however, cannot be released under any other type of license. The
requirement for fonts to remain under this license does not apply
to any document created using the fonts or their derivatives.

  .

  DEFINITIONS
  "Font Software" refers to the set of files released by the Copyright
Holder(s) under this license and clearly marked as such. This may
include source files, build scripts and documentation.

  .

  "Reserved Font Name" refers to any names specified as such after the
copyright statement(s).

  .

  "Original Version" refers to the collection of Font Software components as
distributed by the Copyright Holder(s).

  .

  "Modified Version" refers to any derivative made by adding to, deleting,
or substituting -- in part or in whole -- any of the components of the
Original Version, by changing formats or by porting the Font Software to a
new environment.

  .

  "Author" refers to any designer, engineer, programmer, technical
writer or other person who contributed to the Font Software.

  .

  PERMISSION & CONDITIONS
  Permission is hereby granted, free of charge, to any person obtaining
a copy of the Font Software, to use, study, copy, merge, embed, modify,
redistribute, and sell modified and unmodified copies of the Font
Software, subject to the following conditions:
1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

1.1354 failureaccess 1.0.1
1.355 orc-shims 1.5.1

Available under license:

ORC Shims
Copyright 2013-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1356 methods 1.1.2 1.1.2

1.1356.1 Available under license :
(The MIT License)

Copyright (c) 2013-2014 TJ Holowaychuk <t@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
1.1357 api-spec-converter 2.12.0 2.12.2

1.1357.1 Available under license:

core-js@2.5.1
MIT
Copyright (c) 2014-2017 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

webpack@3.6.0
MIT
Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

zone.js@0.8.18
MIT
The MIT License

Copyright (c) 2016 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

@angular/core@4.4.5
MIT
MIT

@angular/router@4.4.5
MIT
MIT

@angular/http@4.4.5
MIT
MIT

@angular/forms@4.4.5
MIT
MIT

@angular/platform-browser@4.4.5
MIT
MIT
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

webpack@3.11.0
MIT
Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

bootstrap-sass@3.4.1
MIT
The MIT License (MIT)

Copyright (c) 2011-2016 Twitter, Inc
Copyright (c) 2011-2016 The Bootstrap Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

@angular/core@4.4.7
MIT
MIT

@angular/platform-browser@4.4.7
MIT
MIT

@angular/router@4.4.7
MIT
MIT

@angular/forms@4.4.7
MIT
MIT

@angular/http@4.4.7
MIT
MIT

@angular/common@4.4.7
MIT
MIT

file-saver@1.3.8
MIT
The MIT License

Copyright 2016 [Eli Grey][1].

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1358 apache-log4j 2.14.0

1.1358.1 Available under license :

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
Open Source Used In DNAC Platform Arica 5016

See the License for the specific language governing permissions and limitations under the License.

1.1359 asm-tree 7.3.1

1.1359.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.
/**
 * Constructs a new {@link RecordComponentNode}. <i>Subclasses must not use this constructor</i>
 * Instead, they must use the {@link #RecordComponentNode(int, int, String, String, String)}
 * version.
 *
 * @param access the record component access flags (see {@link org.objectweb.asm.Opcodes}). The
 * only valid value is {@link Opcodes#ACC_DEPRECATED}.
 * @param name the record component name.
 * @param descriptor the record component descriptor (see {@link org.objectweb.asm.Type}).
 * @param signature the record component signature.
 * @throws IllegalStateException If a subclass calls this constructor.
 * @deprecated this API is experimental.
 */

Found in path(s):
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1-
  jar/org/objectweb/asmtree/RecordComponentNode.java

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
//    notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
//    notice, this list of conditions and the following disclaimer in the
//    documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
//    this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/FieldInsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/LineNumberNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/ParameterNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/LocalVariableAnnotationNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/InvokeDynamicInsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/InsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/TypeInsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/LdcInsnNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/TryCatchBlockNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/LabelNode.java
* /opt/cola/permits/1135907466_1613628634.84/0/asm-tree-7-3-1-sources-1.jar/org/objectweb/asm/tree/MultiANewArrayInsnNode.java

Open Source Used In DNAC Platform Arica  5017
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
* Redistribution and use in source and binary forms, with or without
  modification, are permitted provided that the following conditions
  are met:
  1. Redistributions of source code must retain the above copyright
     notice, this list of conditions and the following disclaimer.
  2. Redistributions in binary form must reproduce the above copyright
     notice, this list of conditions and the following disclaimer in the
     documentation and/or other materials provided with the distribution.
  3. Neither the name of the copyright holders nor the names of its
     contributors may be used to endorse or promote products derived from
     this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
  IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
  ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
  LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
  CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
  SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
  INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
Copyright (c) 1998-2008 Greg Roelofs. All rights reserved.

This software is provided "as is," without warranty of any kind, express or implied. In no event shall the author or contributors be held liable for any damages arising in any way from the use of this software.

The contents of this file are DUAL-LICENSED. You may modify and/or redistribute this software according to the terms of one of the following two licenses (at your option):

LICENSE 1 ("BSD-like with advertising clause"):

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. Redistributions of source code must retain the above copyright notice, disclaimer, and this list of conditions.
2. Redistributions in binary form must reproduce the above copyright notice, disclaimer, and this list of conditions in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   This product includes software developed by Greg Roelofs and contributors for the book, "PNG: The Definitive Guide," published by O'Reilly and Associates.

LICENSE 2 (GNU GPL v2 or later):

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software Foundation,
Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

pnm2png / png2pnm --- conversion from PBM/PGM/PPM-file to PNG-file
copyright (C) 1999-2019 by Willem van Schaik <willem at schaik dot com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

The software is provided "as is", without warranty of any kind, express or
implied, including but not limited to the warranties of merchantability,
fitness for a particular purpose and noninfringement. In no event shall the
authors or copyright holders be liable for any claim, damages or other
liability, whether in an action of contract, tort or otherwise, arising from,
out of or in connection with the software or the use or other dealings in the
software.
COPYRIGHT NOTICE, DISCLAIMER, and LICENSE
=========================================

PNG Reference Library License version 2
=========================================

* Copyright (c) 1995-2019 The PNG Reference Library Authors.
* Copyright (c) 2018-2019 Cosmin Truta.
* Copyright (c) 1996-1997 Andreas Dilger.
* Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.
The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabuddhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of
the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

    Tom Lane
    Glenn Randers-Pehrson
    Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

    John Bowler
    Kevin Bracey
    Sam Bushell
    Magnus Holmgren
    Greg Roelofs
    Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

    Andreas Dilger
    Dave Martindale
    Guy Eric Schalnat
    Paul Schmidt
    Tim Wegner
The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any 
patents or other property right claims or to contest validity of any 
such claims; this section has the sole purpose of protecting the 
integrity of the free software distribution system, which is 
implemented by public license practices. Many people have made 
generous contributions to the wide range of software distributed 
through that system in reliance on consistent application of that 
system; it is up to the author/donor to decide if he or she is willing 
to distribute software through any other system and a licensee cannot 
impose that choice.

This section is intended to make thoroughly clear what is believed to 
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in 
certain countries either by patents or by copyrighted interfaces, the 
original copyright holder who places the Program under this License 
may add an explicit geographical distribution limitation excluding 
those countries, so that distribution is permitted only in or among 
countries not thus excluded. In such case, this License incorporates 
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions 
of the General Public License from time to time. Such new versions will 
be similar in spirit to the present version, but may differ in detail to 
address new problems or concerns.

Each version is given a distinguishing version number. If the Program 
specifies a version number of this License which applies to it and "any 
later version", you have the option of following the terms and conditions 
either of that version or of any later version published by the Free 
Software Foundation. If the Program does not specify a version number of 
this License, you may choose any version ever published by the Free Software 
Foundation.

10. If you wish to incorporate parts of the Program into other free 
programs whose distribution conditions are different, write to the author 
to ask for permission. For software which is copyrighted by the Free 
Software Foundation, write to the Free Software Foundation; we sometimes 
make exceptions for this. Our decision will be guided by the two goals 
of preserving the free status of all derivatives of our free software and 
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY 
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN 
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED 
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF 
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS 
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE 
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, 
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING 
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR 
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, 
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING 
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED 
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY 
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER 
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE 
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest 
possible use to the public, the best way to achieve this is to make it 
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest 
to attach them to the start of each source file to most effectively 
convey the exclusion of warranty; and each file should have at least 
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify 
it under the terms of the GNU General Public License as published by 
the Free Software Foundation; either version 2 of the License, or 
(at your option) any later version.

This program is distributed in the hope that it will be useful, 
but WITHOUT ANY WARRANTY; without even the implied warranty of 
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the 
GNU General Public License for more details.

You should have received a copy of the GNU General Public License 
along with this program; if not, write to the Free Software 
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.1361 @babel/plugin-transform-parameters

7.16.3

1.1361.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1362 foodmart-data-json 0.4

1.1362.1 Available under license:

foodmart-data-json
Foodmart data set in JSON format
Copyright (C) 2013-2013 Pentaho
Copyright (C) 2015-2015 Julian Hyde
All Rights Reserved.

Based upon the Pentaho mondrian-data-foodmart-json artifact developed as part of the Pentaho Mondrian OLAP engine (http://mondrian.pentaho.com).

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this software except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1363 metrics-jvm 4.0.2

1.1364 js-base64 2.4.8 2.4.8
1.1364.1 Available under license:

Copyright (c) 2014, Dan Kogai
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of {{project}} nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1365 bluebird 3.7.2 3.7.2
1.1365.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013-2018 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1366 require-uncached 1.0.3
1.1366.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1367 xml-beans 4.0.0
1.1367.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

========================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License,   ==
==  Version 2.0, in this case for the Apache XmlBeans distribution.   ==
========================================================================

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:

Aside from contributions to the Apache XMLBeans project, this software also includes:

- one or more source files from the Apache Xerces-J and Apache Axis products, Copyright (c) 1999-2003 Apache Software Foundation

- W3C XML Schema documents Copyright 2001-2003 (c) World Wide Web Consortium (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University)

- resolver.jar from Apache Xml Commons project,
  Copyright (c) 2001-2003 Apache Software Foundation

1.1368 object-component 0.0.3 0.0.3
1.1368.1 Available under license :
The MIT License (MIT)
Copyright (c) 2012 Mike Ihbe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
1.1369 jetty-security 9.4.34.v20201102

1.1369.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and
consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

===================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
===================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.
Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with
classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1370 shebang-command 1.2.0 1.2.0
1.1370.1 Available under license:
The MIT License (MIT)

Copyright (c) Kevin Martensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1371 table 3.8.3
1.1371.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1372 browserslist 3.2.8

1.1372.1 Available under license:

The MIT License (MIT)

Copyright 2014 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1373 latencyutils 2.0.3

1.1374 is-accessor-descriptor 1.0.0 1.0.0

1.1374.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1375 diff-match-patch 1.0.5 1.0.5

1.1375.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1376 javax-ws-rs-api 2.0.1

1.1376.1 Available under license:

Found license 'General Public License 2.0' in "* Copyright (c) 2011-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2012-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2012-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2012-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2012-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in "* Copyright (c) 2010-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2
1.1377 jersey-container-servlet-core 2.25.1

1.1377.1 Available under license:

CDDL+GPL License

http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1378 drange 1.1.1 1.1.1

1.1378.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 David Tudury

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1379 jnr-constants 0.9.0

1.1380 parse-passwd 1.0.0 1.0.0
1.1380.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Brian Woodward

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1381 async 1.5.2
1.1381.1 Available under license:

The MIT License (MIT)

Copyright (c) Spatie bvba <info@spatie.be>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1382 netty-tcnative 2.0.1.Final
1.1382.1 Available under license:
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bundle-Description: A Mavenized fork of Tomcat Native which incorporates various patches. This artifact is dynamically linked to OpenSSL and Apache APR.
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0
Bundle-SymbolicName: io.netty.tcnative
Archiver-Version: Plexus Archiver
Built-By: norman
Bnd-LastModified: 1492498771597
Bundle-ManifestVersion: 2
Bundle-DocURL: http://netty.io/
Bundle-Vendor: The Netty Project
Tool: Bnd-2.1.0.20130426-122213
Export-Package: io.netty.internal.tcnative;version="2.0.1.Final"
Bundle-Name: Netty/TomcatNative [OpenSSL - Dynamic]
Bundle-Version: 2.0.1.Final
Bundle-NativeCode: META-INF/native/libnetty-tcnative.jnilib;osname=macos;osname=macosx;processor=x86_64
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_121

Found in path(s):
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources-jar/META-INF/MANIFEST.MF
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
 * /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources.jar/io/netty/internal/tcnative/Buffer.java
 * /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources.jar/io/netty/internal/tcnative/SSL.java
 * /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources.jar/io/netty/internal/tcnative/SSLContext.java
 * /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources.jar/io/netty/internal/tcnative/Library.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 */
The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

*/

Found in path(s):
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tnative-2-0-1-final-sources-
jar/io/netty/internal/tnative/SessionTicketKey.java
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tnative-2-0-1-final-sources-
jar/io/netty/internal/tnative/CertificateRequestedCallback.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tnative-2-0-1-final-sources-
jar/io/netty/internal/tnative/CertificateVerifier.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources-
  jar/io/netty/internal/tcnative/NativeStaticallyReferencedJniMethods.java
* /opt/cola/permits/1202446267_1631568188.82/0/netty-tcnative-2-0-1-final-sources-
  jar/io/netty/internal/tcnative/SniHostNameMatcher.java

1.1383 swagger-methods 1.0.8 1.0.8
1.1383.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 James Messinger

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1384 tar-stream 1.6.2 1.6.2
1.1384.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Mathias Buus
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1385 jsr305 3.0.2
1.1385.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 */

Found in path(s):
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/GuardedBy.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/ThreadSafe.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/Immutable.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/NotThreadSafe.java

1.1386 babel-plugin-transform-property-literals 7.16.0
1.1386.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1387 static-eval 2.0.2 2.0.2

1.1387.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1388 core-js 3.19.1
1.1388.1 Available under license :

Copyright (c) 2014-2021 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1389 msexcel-builder-extended 0.0.8 0.0.8
1.1389.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 Charlie Zheng

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1390 babel-plugin-syntax-trailing-function-commas 6.22.0 6.22.0

1.1390.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1391 influx 5.0.7 5.0.7

1.1391.1 Available under license :

MIT License

Copyright (c) 2016 Connor Peet

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1392 decamelize 1.2.0 1.2.0
1.1392.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1393 output-file-sync 1.1.2
1.1393.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 - 2016 Shinnosuke Watanabe

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1394 streamsearch 0.1.2 0.1.2
1.1394.1 Available under license:
Copyright Brian White. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1395 stream-to-array 2.3.0 2.3.0
1.1395.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1396 buffer-more-ints 1.0.0 1.0.0

1.1396.1 Available under license :
Copyright (c) 2012 David Wragg. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1397 find-cache-dir 2.1.0

1.1397.1 Available under license :
MIT License

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1398 jtoml 1.0.0

1.1398.1 Available under license:

Copyright (c) 2013 Alexandre Grison

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1399 jakarta-ws-rs-api 2.1.6

1.1399.1 Available under license:

# Notices for the Jakarta RESTful Web Services Project

This content is produced and maintained by the **Jakarta RESTful Web Services** project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaxrs

## Trademarks

**Jakarta RESTful Web Services** is a trademark of the Eclipse Foundation.
## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxrs-api

## Third-party Content

This project leverages the following third party content.

javaee-api (7.0)

* License: Apache-2.0 AND W3C

JUnit (4.11)

* License: Common Public License 1.0

Mockito (2.16.0)

* Project: http://site.mockito.org
* Source: https://github.com/mockito/mockito/releases/tag/v2.16.0

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program.
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient’s receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS"
BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF
TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR
PURPOSE. Each Recipient is solely responsible for determining the
appropriateness of using and distributing the Program and assumes all
risks associated with its exercise of rights under this Agreement,
including but not limited to the risks and costs of program errors,
compliance with applicable laws, damage to or loss of data, programs
or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS
SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this Agreement, and without further
action by the parties hereto, such provision shall be reformed to the
minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity
(including a cross-claim or counterclaim in a lawsuit) alleging that the
Program itself (excluding combinations of the Program with other software
or hardware) infringes such Recipient's patent(s), then such Recipient's
rights granted under Section 2(b) shall terminate as of the date such
litigation is filed.

All Recipient's rights under this Agreement shall terminate if it
fails to comply with any of the material terms or conditions of this
Agreement and does not cure such failure in a reasonable period of
time after becoming aware of such noncompliance. If all Recipient's
rights under this Agreement terminate, Recipient agrees to cease use
and distribution of the Program as soon as reasonably practicable.
However, Recipient's obligations under this Agreement and any licenses
granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement,
but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

### The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and
any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w`. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c` for details.

The hypothetical commands `show w` and `show c` should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w` and `show c`; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision` (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

**1.1400 p-cancelable 1.1.0**

**1.1400.1 Available under license:**

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.1401 has-yarn 2.1.0**

**1.1401.1 Available under license:**

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1402 szmarczak-http-timer 1.1.2
1.1402.1 Available under license :
MIT License

Copyright (c) 2018 Szymon Marczak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1403 send 0.16.2 0.16.2
1.1403.1 Available under license :
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk
Copyright (c) 2014-2016 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1404 es6-map 0.1.5

1.1404.1 Available under license:

Copyright (C) 2013 Mariusz Nowak (www.medikoo.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1405 setprototypeof 1.0.3 1.0.3
1.1405.1 Available under license:
Copyright (c) 2015, Wes Todd

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1406 fromentries 1.3.2
1.1406.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1407 forever-agent 0.6.1 0.6.1
1.1407.1 Available under license:
Apache License

Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and

   You must cause any modified files to carry prominent notices stating that You changed the files; and

   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1408 babel-plugin-transform-es2015-arrow-functions 6.22.0

1.1408.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software">> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1409 jacoco-agent 0.7.9
1.1409.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0'
Found license 'Eclipse Public License 1.0' in 'under the terms and conditions of the Eclipse Public License Version 1.0 which is subject to the terms and conditions of the following license: All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of the copyright holders nor the names of its this software without specific prior written permission.'
1.1410 babel-plugin-transform-es2015-block-scoped-functions 6.22.0 6.22.0

1.1410.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1411 wrap-ansi 5.1.0 5.1.0

1.1411.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1412 xmlbuilder 9.0.4 9.0.4

1.1412.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013 Ozgur Ozcitak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1413 object-visit 1.0.1 1.0.1

1.1413.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1414 jetty-webapp 9.4.41.v20210516

1.1414.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties of title and non-infringement, and implied warranties of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both
* The Apache 2.0 License
   http://www.apache.org/licenses/LICENSE-2.0.html

   and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following
artifacts may be included in the distribution or downloaded when ALPN
module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications
are hosted at github and both modified and original are under GPL v2 with
classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1415 commons-io 2.11.0
1.1415.1 Available under license:
Apache Commons IO
Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1416 oro 2.0.8

1.1416.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * 
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2000 The Apache Software Foundation. All rights reserved.
 * */

Open Source Used In DNAC Platform Arica 5120
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*       "This product includes software developed by the
*        Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/
No license file was found, but licenses were detected in source scan.

/*
* 
* ====================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* ====================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /*
*
* Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/prefixExample.java
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/awk/prefixExample.java
* No license file was found, but licenses were detected in source scan.
*/
*
* $Id: didNotMatch.java,v 1.7 2003/11/07 20:16:23 dfs Exp $
* /
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/didNotMatch.java
No license file was found, but licenses were detected in source scan.

/*
* *
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* */
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.

* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
  WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
  ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  SUCH DAMAGE.

*===================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/matchResultExample.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: strings.java,v 1.3 2003/08/12 18:11:30 dfs Exp $
The Apache Software License, Version 1.1

Copyright (c) 2002 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/awk/strings.java

No license file was found, but licenses were detected in source scan.

/*
 *
 * ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 *
 * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 * "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

*=====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/Perl5Matcher.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: GlobCompiler.java,v 1.8 2003/11/07 20:16:24 dfs Exp $
*
*=====================================================================

* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/GlobCompiler.java
No license file was found, but licenses were detected in source scan.

/#
* 
* =========================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.

* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
  WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
  ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  SUCH DAMAGE.

*====================================================================*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
No license file was found, but licenses were detected in source scan.

/*
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 * name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
*/
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/MatchActionProcessor.java
No license file was found, but licenses were detected in source scan.

/*
* *
* ====================================================================
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in theirname, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIEDWARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIESOF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AREDISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION ORITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOTLIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OFUSE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED ANDON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUTOF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OFSUCH DAMAGE.

This software consists of voluntary contributions made by manyindividuals on behalf of the Apache Software Foundation. For moreinformation on the Apache Software Foundation, please see

No license file was found, but licenses were detected in source scan.

/*$Id: StringSubstitution.java,v 1.8 2003/11/07 20:16:25 dfs Exp $*/

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rightsreserved.

Redistribution and use in source and binary forms, with or withoutmodification, are permitted provided that the following conditionsare met:

1. Redistributions of source code must retain the above copyrightnotice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyrightnotice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/StringSubstitution.java

No license file was found, but licenses were detected in source scan.

/*
* */

* ===========================================================================
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
*   "This product includes software developed by the
*   Apache Software Foundation (http://www.apache.org/)."
*   Alternately, this acknowledgment may appear in the software itself,
*   if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*===================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/jdfix.java

No license file was found, but licenses were detected in source scan.

/*
*
* ==============================================================
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/oro-2-0-8-sources-
jar/org/apache/oro/text/awk/AwkStreamInput.java
No license file was found, but licenses were detected in source scan.

/*
* *
* ====================================================================
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* */
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/PlusNode.java
No license file was found, but licenses were detected in source scan.

/*
* *
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.
*
* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
  jar/org/apache/oro/text/PatternCacheRandom.java
No license file was found, but licenses were detected in source scan.

/*
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Found in path(s):
/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/MalformedCachePatternException.java
No license file was found, but licenses were detected in source scan.

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
  jar/org/apache/oro/text/awk/AwkCompiler.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: printPasswd.java,v 1.8 2003/11/07 20:16:23 dfs Exp $
* 
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* ==========================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/printPasswd.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: oroToApache.java,v 1.7 2003/11/07 20:16:26 dfs Exp $
* 
* ====================================================================
* The Apache Software License, Version 1.1
*/
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/tools/oroToApache.java

No license file was found, but licenses were detected in source scan.

/*
* *
* ===========================================================================
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* *
*/

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

Found in path(s):

/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/regex/Substitution.java
No license file was found, but licenses were detected in source scan.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
must not be used to endorse or promote products derived from this
software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

/*

No license file was found, but licenses were detected in source scan.

/ *
* 
* The Apache Software License, Version 1.1 *
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights *
* reserved. *
* 
* Redistribution and use in source and binary forms, with or without *
* modification, are permitted provided that the following conditions *
* are met: *
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/perl/Perl5Util.java
No license file was found, but licenses were detected in source scan.

/*
The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

No license file was found, but licenses were detected in source scan.

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their...
name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

No license file was found, but licenses were detected in source scan.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.
3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED `AS IS` AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights
reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/StarNode.java
No license file was found, but licenses were detected in source scan.

/*
 * 
 * ====================================================================
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * 
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 * 
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 * 
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 * 
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
  jar/org/apache/oro/text/perl/ParsedSubstitutionEntry.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: SyntaxTree.java,v 1.8 2003/11/07 20:16:24 dfs Exp $
* *
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/SyntaxTree.java

No license file was found, but licenses were detected in source scan.

/*
The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/io/Perl5FilenameFilter.java
No license file was found, but licenses were detected in source scan.

/*
* *
* ==========================================================================
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* *
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
* *
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
must not be used to endorse or promote products derived from this
software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
special, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
use, data, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
or TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

This file was found, but licenses were detected in source scan.

$/Id: CatNode.java,v 1.7 2003/11/07 20:16:24 dfs Exp $
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Found in path(s):

*/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/CatNode.java*
No license file was found, but licenses were detected in source scan.

/*
 * 
 * ====================================================================
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 * 
 * Redistributions and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * 
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 * 
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 * 
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 * name, without prior written permission of the Apache Software Foundation.
 * 
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/util/CacheFIFO.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: MatcherDemoApplet.java,v 1.5 2003/11/07 20:16:23 dfs Exp $
 *
 * ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 */
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
*===============================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/MatcherDemoApplet.java
No license file was found, but licenses were detected in source scan.

/
* 
* ==========================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.

Open Source Used In DNAC Platform Arica 5165
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* 
* =========================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
/*

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/TokenNode.java
No license file was found, but licenses were detected in source scan.

*/

*
* ====================================================================
* The Apache Software License, Version 1.1
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE Apache SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/io/AwkFilenameFilter.java
No license file was found, but licenses were detected in source scan.

/*
* /
*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*
must not be used to endorse or promote products derived from this
software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

*/

Found in path(s):
/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/Perl5Substitution.java
No license file was found, but licenses were detected in source scan.

*/


The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights
reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* */

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/util/GenericCache.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: groups.java,v 1.8 2003/11/07 20:16:23 dfs Exp $
* * *
* * *
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* *
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* *
* ====================================================================
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

No license file was found, but licenses were detected in source scan.

The Apache Software License, Version 1.1

Copyright (c) 2000-2002 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

No license file was found, but licenses were detected in source scan.

/*
 *
 * ===========================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/).
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* */

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/util/CacheLRU.java

No license file was found, but licenses were detected in source scan.

/*
* 
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.

* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
  WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
  ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  SUCH DAMAGE.

* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/PatternCacheLRU.java
No license file was found, but licenses were detected in source scan.

/**
 *
 * ==========================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ==============================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
path:/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/PatternMatcherInput.java
No license file was found, but licenses were detected in source scan.

/*
 *
 * ==============================================================
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* */

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/regex/Perl5Debug.java
No license file was found, but licenses were detected in source scan.

/*
* 
* ====================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
*
* distribution.

* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/DefaultMatchAction.java
No license file was found, but licenses were detected in source scan.

/*
* *
*ReflectionQL License
* The Apache Software License, Version 1.1
Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/).
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/PatternCacheFIFO.java
No license file was found, but licenses were detected in source scan.

/*
 *
 * ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation.  All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 * name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/PatternMatcher.java
No license file was found, but licenses were detected in source scan.

/*
* *
* The Apache Software License, Version 1.1
* *
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* *
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
*/
Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
must not be used to endorse or promote products derived from this
software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

/}

No license file was found, but licenses were detected in source scan.

/*
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* 
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/util/GenericCacheEntry.java

No license file was found, but licenses were detected in source scan.

/*
 *
 * ==============================================================
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED `AS IS' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,  
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT  
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF  
* SUCH DAMAGE.
*  
* This software consists of voluntary contributions made by many  
* individuals on behalf of the Apache Software Foundation. For more  
* information on the Apache Software Foundation, please see  
* */

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/awk/QuestionNode.java
No license file was found, but licenses were detected in source scan.

/*
 *  
 *====================================================================
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *  
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *  
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *  
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *  
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 
* permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
  jar/org/apache/oro/text/regex/Perl5Compiler.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: OrNode.java,v 1.7 2003/11/07 20:16:24 dfs Exp $
 *
 *===================================================================
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
 *
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
*/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/OrNode.java

No license file was found, but licenses were detected in source scan.

/*
* 
*====================================================================
* The Apache Software License, Version 1.1

* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.

* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/
/**
* Determines if the contents of a PatternMatcherInput, starting from the
* current offset of the input contains a pattern.
* If a pattern match is found, a MatchResult
* instance representing the first such match is made accessible via
* {@link #getMatch()}. The current offset of the
* PatternMatcherInput is set to the offset corresponding to the end
* of the match, so that a subsequent call to this method will continue
* searching where the last call left off. You should remember that the
* region between the begin and end offsets of the PatternMatcherInput are
* considered the input to be searched, and that the current offset
* of the PatternMatcherInput reflects where a search will start from.
* Matches extending beyond the end offset of the PatternMatcherInput
* will not be matched. In other words, a match must occur entirely
* between the begin and end offsets of the input. See
* {@link org.apache.oro.text.regex.PatternMatcherInput PatternMatcherInput}
* for more details.
* As a side effect, if a match is found, the PatternMatcherInput match
* offset information is updated. See the PatternMatcherInput
* {@link org.apache.oro.text.regex.PatternMatcherInput#setMatchOffsets
* setMatchOffsets(int, int)} method for more details.
* The pattern must be an AwkPattern instance, otherwise a
* ClassCastException will be thrown. You are not required to, and
* indeed should NOT try to (for performance reasons), catch a
* ClassCastException because it will never be thrown as long as you use
* an AwkPattern as the pattern parameter.
* This method is usually used in a loop as follows:
* PatternMatcher matcher;
* PatternCompiler compiler;
* Pattern pattern;
* PatternMatcherInput input;
* MatchResult result;
* compiler = new AwkCompiler();
* matcher = new AwkMatcher();
* try {
* pattern = compiler.compile(somePatternString);
* } catch(MalformedPatternException e) {
* System.err.println("Bad pattern.");
*
* System.err.println(e.getMessage());
* return;
* }
*
* input   = new PatternMatcherInput(someStringInput);
* 
* while(matcher.contains(input, pattern)) {
*   result = matcher.getMatch();
*   // Perform whatever processing on the result you want.
* }
*
* </pre></blockquote>
* @param input  The PatternMatcherInput to test for a match.
* @param pattern  The Pattern to be matched.
* @return True if the input contains a pattern match, false otherwise.
* @exception ClassCastException If a Pattern instance other than an
* AwkPattern is passed as the pattern parameter.
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/AwkMatcher.java
No license file was found, but licenses were detected in source scan.

/*
* 
* ====================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/).
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* =========================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/addCommas.java
No license file was found, but licenses were detected in source scan.

/*
* 
* ====================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.

* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
  WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
  ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  SUCH DAMAGE.

* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
/*
 *
 * ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation.  All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED.  IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* */

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/GenericPatternCache.java

No license file was found, but licenses were detected in source scan.

/*
* $Id: NegativeCharacterClassNode.java,v 1.7 2003/11/07 20:16:24 dfs Exp $
* *
* ==============================================================
*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written

permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in theirname, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIEDWARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIESOF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AREDISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION ORITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOTLIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OFUSE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED ANDON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUTOF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OFSUCH DAMAGE.

This software consists of voluntary contributions made by manyindividuals on behalf of the Apache Software Foundation. For moreinformation on the Apache Software Foundation, please see<http://www.apache.org/>.

This file was found, but licenses were detected in source scan.


The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rightsreserved.

Redistribution and use in source and binary forms, with or withoutmodification, are permitted provided that the following conditionsare met:

1. Redistributions of source code must retain the above copyrightnotice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/awk/streamInputExample.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: CharStringPointer.java,v 1.7 2003/11/07 20:16:25 dfs Exp $
 */

* vim: sw=4:
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
  jar/org/apache/oro/text/regex/CharStringPointer.java

No license file was found, but licenses were detected in source scan.

/*
*
* ====================================================================
* The Apache Software License, Version 1.1
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* * Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* *
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
* *
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/grep.java
No license file was found, but licenses were detected in source scan.

/*
* $Id: substituteExample.java,v 1.8 2003/11/07 20:16:23 dfs Exp $
* *
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* *
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*       "This product includes software developed by the
*       Apache Software Foundation (http://www.apache.org)."
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

/*

Found in path(s):
*/

No license file was found, but licenses were detected in source scan.

/*
* $Id: matchesContainsExample.java,v 1.7 2003/11/07 20:16:23 dfs Exp $
*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights reserved.
* Redistirbution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
*/
* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.
*
* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
Alternately, this acknowledgment may appear in the software itself,
if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/awk/matchesContainsExample.java
No license file was found, but licenses were detected in source scan.

/*
*
* =========================================================================
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* *
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* *
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* *
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
* *
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED.  IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/Perl5Pattern.java

No license file was found, but licenses were detected in source scan.

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights
reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

3. The end-user documentation included with the redistribution,
if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

This license file was found, but licenses were detected in source scan.

$Id: substituteExample.java,v 1.9 2003/11/07 20:16:23 dfs Exp $

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights
reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED \"AS IS\" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*===================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
  */

Found in path(s):
  * /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/util/CacheFIFO2.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: splitExample.java,v 1.8 2003/11/07 20:16:23 dfs Exp $
 *
 * ====================================================================
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 * *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 *    must not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 * *
 * 5. Products derived from this software may not be called "Apache"
 *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 *    name, without prior written permission of the Apache Software Foundation.
 * *
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

* DISCLAIMER. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* ================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/splitExample.java
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/awk/splitExample.java

No license file was found, but licenses were detected in source scan.

/*
* 
* ================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* /

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
jar/org/apache/oro/text/regex/PatternCompiler.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: CharacterClassNode.java,v 1.7 2003/11/07 20:16:24 dfs Exp $
 * 
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:

Open Source Used In DNAC Platform Arica 5211
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
   "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache" or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Found in path(s):
  /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/CharacterClassNode.java
/*
 * 
 * /******************************************************************************/
 * The Apache Software License, Version 1.1
 * 
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 * 
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 * 
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 * 
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 * 
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 * name, without prior written permission of the Apache Software Foundation.
 * 
 * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/LeafNode.java
No license file was found, but licenses were detected in source scan.

/*
 * $Id: MalformedPerl5PatternException.java,v 1.7 2003/11/07 20:16:25 dfs Exp $
 *
 * ====================================================================

 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 */
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
*=====================================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-
* jar/org/apache/oro/text/perl/MalformedPerl5PatternException.java
No license file was found, but licenses were detected in source scan.

/*
* *
*=====================================================================
*
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* *
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* *
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
   "This product includes software developed by the
   Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
  WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
  OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
  ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  SUCH DAMAGE.
* ==========================================================================
* The Apache Software License, Version 1.1
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/util/CacheRandom.java
No license file was found, but licenses were detected in source scan.

/*
* 
* The Apache Software License, Version 1.1
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ==============================================================
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
**<http://www.apache.org/>.
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/examples/semicolon.java

No license file was found, but licenses were detected in source scan.

*/
* $Id: MalformedPatternException.java,v 1.8 2003/11/07 20:16:25 dfs Exp $
*
* ====================================================================
* The Apache Software License, Version 1.1
* Copyright (c) 2000 The Apache Software Foundation. All rights reserved.
* Redistributions and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*       "This product includes software developed by the
*       Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR

Open Source Used In DNAC Platform Arica 5218
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
  * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
  * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
  * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
  * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
  * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
  * SUCH DAMAGE.
  *
  * This software consists of voluntary contributions made by many
  * individuals on behalf of the Apache Software Foundation. For more
  * information on the Apache Software Foundation, please see
  */

Found in path(s):
*/opt/cola/permits/1000740375_1646171523.75/oro-2-0-8-sources-jar/org/apache/oro/text/regex/MalformedPatternException.java

No license file was found, but licenses were detected in source scan.

/*
 *
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache"
   or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

This software consists of voluntary contributions made by many
individuals on behalf of the Apache Software Foundation. For more
information on the Apache Software Foundation, please see

Found in path(s):
/opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/regex/MatchResult.java
No license file was found, but licenses were detected in source scan.

/*
* 
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*    
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*    
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
*    
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
*    
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.
*    
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* 
* ====================================================================
* 
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* 
* Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/MatchAction.java
* No license file was found, but licenses were detected in source scan.
*/
* 
* ====================================================================
* The Apache Software License, Version 1.1
* 
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
* 
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
* 
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
* 
* 5. Products derived from this software may not be called "Apache"
* or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
* name, without prior written permission of the Apache Software Foundation.
* 
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* ====================================================================
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/

Found in path(s):
* /opt/cola/permits/1000740375_1646171523.75/0/oro-2-0-8-sources-jar/org/apache/oro/text/awk/AwkPattern.java

1.1417 qt 5.7.0
1.1417.1 Available under license:
libjpeg-turbo Licenses

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

- The IJG (Independent JPEG Group) License, which is listed in
  [README.ijg](README.ijg)

  This license applies to the libjpeg API library and associated programs
  (any code inherited from libjpeg, and any modifications to that code.)

- The Modified (3-clause) BSD License, which is listed below

  This license covers the TurboJPEG API library and associated programs, as
  well as the build system.

- The zlib License, which is listed below

  This license is a subset of the other two, and it covers the libjpeg-turbo
  SIMD extensions.

Complying with the libjpeg-turbo Licenses

This section provides a roll-up of the libjpeg-turbo licensing terms, to the
best of our understanding.

1. If you are distributing a modified version of the libjpeg-turbo source,
   then:

   1. You cannot alter or remove any existing copyright or license notices
from the source.

**Origin**
- Clause 1 of the IJG License
- Clause 1 of the Modified BSD License
- Clauses 1 and 3 of the zlib License

2. You must add your own copyright notice to the header of each source file you modified, so others can tell that you modified that file (if there is not an existing copyright header in that file, then you can simply add a notice stating that you modified the file.)

**Origin**
- Clause 1 of the IJG License
- Clause 2 of the zlib License

3. You must include the IJG README file, and you must not alter any of the copyright or license text in that file.

**Origin**
- Clause 1 of the IJG License

2. If you are distributing only libjpeg-turbo binaries without the source, or if you are distributing an application that statically links with libjpeg-turbo, then:

1. Your product documentation must include a message stating:

   This software is based in part on the work of the Independent JPEG Group.

   **Origin**
   - Clause 2 of the IJG license

2. If your binary distribution includes or uses the TurboJPEG API, then your product documentation must include the text of the Modified BSD License.

   **Origin**
   - Clause 2 of the Modified BSD License

3. You cannot use the name of the IJG or The libjpeg-turbo Project or the contributors thereof in advertising, publicity, etc.

   **Origin**
   - IJG License
   - Clause 3 of the Modified BSD License
4. The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be free of defects, nor do we accept any liability for undesirable consequences resulting from your use of the software.

**Origin**
- IJG License
- Modified BSD License
- zlib License

The Modified (3-clause) BSD License

Copyright (C) \<YEAR\> \<AUTHOR\>. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS", AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The zlib License

Copyright (C) \<YEAR\>, \<AUTHOR\>.

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.
Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Why Three Licenses?
===================

The zlib License could have been used instead of the Modified (3-clause) BSD License, and since the IJG License effectively subsumes the distribution conditions of the zlib License, this would have effectively placed libjpeg-turbo binary distributions under the IJG License. However, the IJG License specifically refers to the Independent JPEG Group and does not extend attribution and endorsement protections to other entities. Thus, it was desirable to choose a license that granted us the same protections for new code that were granted to the IJG for code derived from their software.

This file is part of the Independent JPEG Group's software.

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1994-1996, Thomas G. Lane. All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:
(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".
(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.
These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

Copyright (c) 2013 Jake Luer <jake@alogicalparadox.com> (http://alogicalparadox.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>
*/

Copyright (C) 2002, 2003 CodeFactory AB
Copyright (C) 2004, 2005 Red Hat, Inc.

Licensed under the Academic Free License version 2.1

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

---

The Academic Free License
v.2.1

This Academic Free License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following notice immediately following the copyright notice for the Original Work:

Licensed under the Academic Free License version 2.1

1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:
   a) to reproduce the Original Work in copies;
   b) to prepare derivative works ("Derivative Works") based upon the Original Work;
   c) to distribute copies of the Original Work and Derivative Works to the public;
   d) to perform the Original Work publicly; and
   e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual,
sublicenseable license, under patent claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.

4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to license.

5) This section intentionally omitted.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries,
and by international treaty. Therefore, by exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, against Licensor or any licensee alleging that the Original Work infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware.

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. 101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys' fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.

13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.

This license is Copyright (C) 2003-2004 Lawrence E. Rosen. All rights reserved. Permission is hereby granted to copy and distribute this license without modification. This license may not be modified without the express written permission of its copyright owner.

---

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:
   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date
      of any change.
   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from
      the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
      License.
   c) If the modified program normally reads commands interactively when run, you must cause it, when started
      running for such interactive use in the most ordinary way, to print or display an announcement including an
      appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
      that users may redistribute the program under these conditions, and telling the user how to view a copy of this
      License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
      work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived
from the Program, and can be reasonably considered independent and separate works in themselves, then this
License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole
must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to
each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based
on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of
this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:
   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed
      under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
      than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
      source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
      interchange; or,
   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This
      alternative is allowed only for noncommercial distribution and only if you received the program in object code or
      executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an
executable work, complete source code means all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation and installation of the executable. However, as
a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program's name and an idea of what it does.>
Copyright (C) <yyyy> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

This source code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

Open Source Used in DNAC Platform Arica 5234
Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989 Ty Coon, President of Vice

MIT License

Copyright (C) 2017 by Marijn Haverbeke <marijnh@gmail.com> and others

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-2018 Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014 Max Ogden and other contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Paul Hsieh OLD BSD license

Copyright (c) 2010, Paul Hsieh
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither my name, Paul Hsieh, nor the names of any other contributors to the code use may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HoldERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright (c) 2009 The RE2 Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
The JsonCpp library's source code, including accompanying documentation, tests and demonstration applications, are licensed under the following conditions...

Baptiste Lepilleur and The JsonCpp Authors explicitly disclaim copyright in all jurisdictions which recognize such a disclaimer. In such jurisdictions, this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of 2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur and The JsonCpp Authors, and is released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this software may choose to accept it either as 1) Public Domain, 2) under the conditions of the MIT License (see below), or 3) under the terms of dual Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

========================================================================
Copyright (c) 2007-2010 Baptiste Lepilleur and The JsonCpp Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

========================================================================
(END LICENSE TEXT)

The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the Public Domain "license" you can re-license your copy using whatever license you like.

Copyright (c) 2015-2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

MODIFICATIONS TO THIS FILE MAY MEAN IT NO LONGER ACCURATELY REFLECTS KHRONOS STANDARDS. THE UNMODIFIED, NORMATIVE VERSIONS OF KHRONOS SPECIFICATIONS AND HEADER INFORMATION ARE LOCATED AT https://www.khronos.org/registry/

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

The MIT License (MIT)

Copyright (c) 2014 Bitfighter developers

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU GENERAL PUBLIC LICENSE
The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU General Public License version 2, which is displayed below.

-------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software -- to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent license
would not permit royalty-free redistribution of the Program by all
those who receive copies directly or indirectly through you, then the
only way you could satisfy both it and this License would be to refrain
entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may
add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the limitation
as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail
to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by
the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by
the Free Software Foundation, write to the Free Software Foundation;
we sometimes make exceptions for this. Our decision will be guided by
the two goals of preserving the free status of all derivatives of our
free software and of promoting the sharing and reuse of software
generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY
MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE
TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Notice that the following BSD-style license applies to the Valgrind header files used by Chromium (valgrind.h and memcheck.h). However, the rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated.
permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2012 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2011 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Open Asset Import Library (assimp)

Copyright (c) 2006-2016, assimp team
All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the assimp team, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission of the assimp team.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
AN EXCEPTION applies to all files in the ./test/models-nonbsd folder. These are 3d models for testing purposes, from various free sources on the internet. They are - unless otherwise stated - copyright of their respective creators, which may impose additional requirements on the use of their work. For any of these models, see <model-name>.source.txt for more legal information. Contact us if you are a copyright holder and believe that we credited you improperly or if you don't want your files to appear in the repository.

Poly2Tri Copyright (c) 2009-2010, Poly2Tri Contributors
http://code.google.com/p/poly2tri/

All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Poly2Tri nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.
We call this license the "Lesser" General Public License because it does less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

   If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

   However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

   When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

   If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

   Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Licence for libxslt except libexslt

------------------------------------------------------------------

Copyright (C) 2001-2002 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

----------------------------------------------------------------------

Licence for libexslt
----------------------------------------------------------------------

Copyright (C) 2001-2002 Thomas Broyer, Charlie Bozeman and Daniel Veillard.
All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the authors shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

----------------------------------------------------------------------

(The MIT License)

Copyright (c) 2014 Nathan Rajlich <nathan@tootallnate.net>

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017, 2018 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2010 Xtreme Labs and Pivotal Labs

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) Kevin Mrtensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----

Most code has one of the following copyrights:

Copyright (C) 2002-2005 3Dlabs Inc. Ltd.
Copyright (C) 2012-2014 LunarG, Inc.
Copyright (C) 2002-2010 The ANGLE Project Authors.
Copyright (C) 2015-2016 Google, Inc.

and is covered under the following license (BSD):

//
//--All rights reserved.
//
//--Redistribution and use in source and binary forms, with or without
//--modification, are permitted provided that the following conditionsA
//--are met:
//--
//-- Redistributions of source code must retain the above copyright
//-- notice, this list of conditions and the following disclaimer.
//--
//-- Redistributions in binary form must reproduce the above
//-- copyright notice, this list of conditions and the following
//-- disclaimer in the documentation and/or other materials provided
//-- with the distribution.
//--
//-- Neither the name of 3Dlabs Inc. Ltd. nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
// FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
// COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
// BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
// LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
// CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
// LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
// ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
// POSSIBILITY OF SUCH DAMAGE.
/

-----

Files under glslang/MachineIndependent/preprocessor used under
the following license (BSD-like):

****************************************************************************
Copyright (c) 2002, NVIDIA Corporation.

NVIDIA Corporation("NVIDIA") supplies this software to you in
consideration of your agreement to the following terms, and your use,
installation, modification or redistribution of this NVIDIA software
constitutes acceptance of these terms. If you do not agree with these
terms, please do not use, install, modify or redistribute this NVIDIA
software.

In consideration of your agreement to abide by the following terms, and
subject to these terms, NVIDIA grants you a personal, non-exclusive
license, under NVIDIA's copyrights in this original NVIDIA software (the
"NVIDIA Software"), to use, reproduce, modify and redistribute the
NVIDIA Software, with or without modifications, in source and/or binary
forms; provided that if you redistribute the NVIDIA Software, you must
retain the copyright notice of NVIDIA, this notice and the following
text and disclaimers in all such redistributions of the NVIDIA Software.
Neither the name, trademarks, service marks nor logos of NVIDIA
Corporation may be used to endorse or promote products derived from the
NVIDIA Software without specific prior written permission from NVIDIA.
Except as expressly stated in this notice, no other rights or licenses
express or implied, are granted by NVIDIA herein, including but not
limited to any patent rights that may be infringed by your derivative
works or by other works in which the NVIDIA Software may be
incorporated. No hardware is licensed hereunder.
THE NVIDIA SOFTWARE IS BEING PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR ITS USE AND OPERATION EITHER ALONE OR IN COMBINATION WITH OTHER PRODUCTS.

IN NO EVENT SHALL NVIDIA BE LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL, EXEMPLARY, CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, LOST PROFITS; PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) OR ARISING IN ANY WAY OUT OF THE USE, REPRODUCTION, MODIFICATION AND/OR DISTRIBUTION OF THE NVIDIA SOFTWARE, HOWEVER CAUSED AND WHETHER UNDER THEORY OF CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY OR OTHERWISE, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The file glslang/MachineIndependent/gl_types.h has the following license (MIT):

/*
 ** Copyright (c) 2013 The Khronos Group Inc.
 **
 ** Permission is hereby granted, free of charge, to any person obtaining a
 ** copy of this software and/or associated documentation files (the
 ** "Materials"), to deal in the Materials without restriction, including
 ** without limitation the rights to use, copy, modify, merge, publish,
 ** distribute, sublicense, and/or sell copies of the Materials, and to
 ** permit persons to whom the Materials are furnished to do so, subject to
 ** the following conditions:
 **
 ** The above copyright notice and this permission notice shall be included
 ** in all copies or substantial portions of the Materials.
 **
 ** THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 ** EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 ** MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
 ** IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
 ** CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 ** TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
 ** MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
 */

The files glslang/SPRV/GLSL.std.450.h and
glslang/SPIRV/spirv.hpp have the following license.

/*
 ** Copyright (c) 2014-2016 The Khronos Group Inc.
 **
 ** Permission is hereby granted, free of charge, to any person obtaining a copy
 ** of this software and/or associated documentation files (the “Materials”),
 ** to deal in the Materials without restriction, including without limitation
 ** the rights to use, copy, modify, publish, distribute, sublicense,
 ** and/or sell copies of the Materials, and to permit persons to whom the
 ** Materials are furnished to do so, subject to the following conditions:
 **
 ** The above copyright notice and this permission notice shall be included in
 ** all copies or substantial portions of the Materials.
 **
 ** MODIFICATIONS TO THIS FILE MAY MEAN IT NO LONGER ACCURATELY REFLECTS KHRONOS
 ** STANDARDS. THE UNMODIFIED, NORMATIVE VERSIONS OF KHRONOS SPECIFICATIONS AND
 ** HEADER INFORMATION ARE LOCATED AT https://www.khronos.org/registry/
 **
 ** THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
 ** OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 ** FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
 ** THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 ** LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
 ** FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS
 ** IN THE MATERIALS.
 */

-----

Files under tools/data contain portions of GNU Bison, and are used under
the following license (GPL v3), with an exception as described at

In particular, when parsers are generated from the template files, an
exception clause is inserted into the generated source files which allows
their use in non-free programs.


# This program is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
Copyright (c) 2010-2017 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013-2017 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Contributions to the collaboration shall not be considered confidential.

Each contributor represents and warrants that it has the right and authority to license copyright in its contributions to the collaboration.

Each contributor agrees to license the copyright in the contributions under the Modified (2-clause or 3-clause) BSD License or the Clear BSD License.

Please see the IPR statements submitted to the IETF for the complete
patent licensing details:

Xiph.Org Foundation:
https://datatracker.ietf.org/ipr/1524/

Microsoft Corporation:
https://datatracker.ietf.org/ipr/1914/

Skype Limited:
https://datatracker.ietf.org/ipr/1602/

Broadcom Corporation:
https://datatracker.ietf.org/ipr/1526/

The MIT License (MIT)

Copyright (c) 2016 crypto-browserify

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# Names should be added to this file as:
# 
Lei Zhang <antiagainst@google.com>
David Neto <dneto@google.com>
Andrew Woloszyn <awoloszyn@google.com>
Stefanus Du Toit <sdt@google.com>
Dejan Mircevski <deki@google.com>
Mark Adams <marka@nvidia.com>
Jason Ekstrand <jason.ekstrand@intel.com>
Damien Mabin <dmabin@google.com>
Qining Lu <qining@google.com>
Jakob Vogel <JakobpunktVogel@gmail.com>
David Yen <dyen@google.com>
Adam Chainz <adam@adamj.eu>

Copyright (c) 2010, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software is released under the MIT license:

Copyright (c) James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2017, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2006-2013 James Graham and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: perfetto
Source: https://android.googlesource.com/platform/external/perfetto/

Files: *
Copyright: Copyright (C) 2017 The Android Open Source Project
License: Apache 2
Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
/*
 * SpanDSP - a series of DSP components for telephony
 */
* g722_decode.c - The ITU G.722 codec, decode part.
* Written by Steve Underwood <steveu@coppice.org>
* Copyright (C) 2005 Steve Underwood
* Despite my general liking of the GPL, I place my own contributions
  to this code in the public domain for the benefit of all mankind -
  even the slimy ones who might try to proprietize my work and use it
  to my detriment.
* Based in part on a single channel G.722 codec which is:
* Copyright (c) CMU 1993
  Computer Science, Speech Group
  Chengxiang Lu and Alex Hauptmann
*/

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2000 Hans Petter Bieker. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2000 by Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2009, 2010, 2013-2016 by the Brotli Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

PCRE2 LICENCE
-------------

PCRE2 is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Releases 10.00 and above of PCRE2 are distributed under the terms of the "BSD" licence, as specified below, with one exemption for certain binary redistributions. The documentation for PCRE2, supplied in the "doc" directory, is distributed under the same terms as the software itself. The data in the testdata directory is not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also included in the distribution is a just-in-time compiler that can be used to optimize pattern matching. This is an optional feature that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS
-----------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2019 University of Cambridge
All rights reserved.

PCRE2 JUST-IN-TIME COMPILATION SUPPORT
----------------------------------------

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu
STACK-LESS JUST-IN-TIME COMPILER

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notices, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notices, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

EXEMPTION FOR BINARY LIBRARY-LIKE PACKAGES

----------------------------------
The second condition in the BSD licence (covering binary redistributions) does not apply all the way down a chain of software. If binary package A includes PCRE2, it must respect the condition, but if package B is software that includes package A, the condition is not imposed on package B unless it uses PCRE2 independently.

End

The MIT License (MIT)

Copyright (c) 2012-2019 Paul Miller (https://paulmillr.com), Elan Shanker

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2012, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
special, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any,
   must include the following acknowledgment:

   "This product includes software developed by the Indiana University
   Extreme! Lab (http://www.extreme.indiana.edu/)."

   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana University" and "Indiana University Extreme! Lab"
   must not be used to endorse or promote products derived from this
   software without prior written permission. For written permission,
   please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University"
   name nor may "Indiana University" appear in their name, without prior
   written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) Kevin Martensson <kevinmartensson@gmail.com> (github.com/kevva)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2011 The Chromium OS Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2014 Kim Kulling

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright  2011-2015 Paul Vorbach <paul@vorba.ch>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the Software), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2016 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of
GNU Lesser General Public License version 3, which is displayed below.
This license makes reference to the version 3 of the GNU General
Public License, which you can find in the LICENSE.GPL3 file.

GNUSMALLER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this
licensedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, this License refers to version 3 of the GNU Lesser
General Public License, and the GNU GPL refers to version 3 of the
GNU General Public License.

The Library refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An Application is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A Combined Work is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the Linked
Version.

The Minimal Corresponding Source for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The Corresponding Application Code for a Combined Work means the
object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:
a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License or any later version applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

(MIT License)

Copyright (c) 2013 [Ramesh Nair](http://www.hiddentao.com/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CMake - Cross Platform Makefile Generator
Copyright 2000-2009 Kitware, Inc., Insight Software Consortium
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of Kitware, Inc., the Insight Software Consortium, nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The above copyright and license notice applies to distributions of CMake in source and binary form. Some source files contain additional notices of original copyright by their contributors; see each source for details. Third-party software packages supplied with CMake under compatible licenses provide their own copyright notices documented in corresponding subdirectories.
CMake was initially developed by Kitware with the following sponsorship:

* National Library of Medicine at the National Institutes of Health as part of the Insight Segmentation and Registration Toolkit (ITK).

* US National Labs (Los Alamos, Livermore, Sandia) ASC Parallel Visualization Initiative.

* National Alliance for Medical Image Computing (NAMIC) is funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149.

* Kitware, Inc.
Copyright 2012-2015 The Dojo Foundation <http://dojofoundation.org/>
Based on Underscore.js, copyright 2009-2015 Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
/
* Copyright 2011 The WebRTC Project Authors. All rights reserved.
*
* Use of this source code is governed by a BSD-style license
* that can be found in the LICENSE file in the root of the source
* tree. An additional intellectual property rights grant can be found
* in the file PATENTS. All contributing project authors may
* be found in the AUTHORS file in the root of the source tree.
*/

Copyright 2012 Square Inc.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright  2002 USC/Information Sciences Institute

Permission to use, copy, modify, distribute, and sell this software
and its documentation for any purpose is hereby granted without
fee, provided that the above copyright notice appear in all copies
and that both that copyright notice and this permission notice
appear in supporting documentation, and that the name of
Information Sciences Institute not be used in advertising or
publicity pertaining to distribution of the software without
specific, written prior permission.  Information Sciences Institute
makes no representations about the suitability of this software for
any purpose.  It is provided "as is" without express or implied
warranty.

INFORMATION SCIENCES INSTITUTE DISCLAIMS ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS.  IN NO EVENT SHALL INFORMATION SCIENCES
INSTITUTE BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA
OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (C) 1995-2002 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com
Copyright (c) 2016 Douglas Christopher Wilson doug@somethingdoug.com

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN

Open Source Used In DNAC Platform Arica 5290
THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright 2017 Kat Marchn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2007, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-------------------------------------------------------------------------------

The file url_parse.cc is based on nsURLParsers.cc from Mozilla. This file is licensed separately as follows:

The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is mozilla.org code.
The Initial Developer of the Original Code is
Netscape Communications Corporation.
Portions created by the Initial Developer are Copyright (C) 1998
the Initial Developer. All Rights Reserved.

Contributor(s):
Darin Fisher (original author)

Alternatively, the contents of this file may be used under the terms of
either the GNU General Public License Version 2 or later (the “GPL”), or
the GNU Lesser General Public License Version 2.1 or later (the “LGPL”),
in which case the provisions of the GPL or the LGPL are applicable instead
of those above. If you wish to allow use of your version of this file only
under the terms of either the GPL or the LGPL, and not to allow others to
use your version of this file under the terms of the MPL, indicate your
decision by deleting the provisions above and replace them with the notice
and other provisions required by the GPL or the LGPL. If you do not delete
the provisions above, a recipient may use your version of this file under
the terms of any one of the MPL, the GPL or the LGPL.
Copyright (c) 2015, Wes Todd

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) Microsoft Corporation. All rights reserved.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use
this file except in compliance with the License. You may obtain a copy of the
License at http://www.apache.org/licenses/LICENSE-2.0

THIS CODE IS PROVIDED ON AN *AS IS* BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED
WARRANTIES OR CONDITIONS OF TITLE, FITNESS FOR A PARTICULAR PURPOSE,
MERCHANTABILITY OR NON-INFRINGEMENT.

See the Apache Version 2.0 License for specific language governing permissions
and limitations under the License.

HarfBuzz was previously licensed under different licenses. This was
changed in January 2008. If you need to relicense your old copies,
consult the announcement of the license change on the internet. Other than that, each copy of HarfBuzz is licensed under the COPYING file included with it. The actual license follows:

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Espree
Copyright JS Foundation and other contributors, https://js.foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Javier Blanco
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2011 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2014 Andrew Kelley

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files
(the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of the Software,
and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2006 Bob Ippolito

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Kevin Gravier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
SOFTWARE.
Copyright (c) 2013, Jim Higson
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the 'Software'), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT
LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (Expat)

Copyright (c) 2014 Andrew Kelley

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files
(the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of the Software,
and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HoldERS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2016 Cory Benfield and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014, Eugene Ware

All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Eugene Ware nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY EUGENE WARE "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL EUGENE WARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------
Parts of jpeg-js-decoder.js is also licensed under Apache-2.0 license. See source file for license.
This license applies to all parts of V8 that are not externally maintained libraries. The externally maintained libraries used by V8 are:

- PCRE test suite, located in test/mjsunit/third_party/regexp-pcre/regexp-pcre.js. This is based on the test suite from PCRE-7.3, which is copyrighted by the University of Cambridge and Google, Inc. The copyright notice and license are embedded in regexp-pcre.js.

- Layout tests, located in test/mjsunit/third_party/object-keys. These are based on layout tests from webkit.org which are copyrighted by Apple Computer, Inc. and released under a 3-clause BSD license.

- Strongtalk assembler, the basis of the files assembler-arm-inl.h, assembler-arm.cc, assembler-arm.h, assembler-ia32-inl.h, assembler-ia32.cc, assembler-ia32.h, assembler-x64-inl.h, assembler-x64.cc, assembler-x64.h, assembler-mips-inl.h, assembler-mips.cc, assembler-mips.h, assembler.cc and assembler.h. This code is copyrighted by Sun Microsystems Inc. and released
under a 3-clause BSD license.

- Valgrind client API header, located at src/third_party/valgrind/valgrind.h
  This is released under the BSD license.

- The Wasm C/C++ API headers, located at third_party/wasm-api/wasm.{h,hh}
  This is released under the Apache license. The API's upstream prototype
  implementation also formed the basis of V8's implementation in
  src/wasm/c-api.cc.

These libraries have their own licenses; we recommend you read them,
as their terms may differ from the terms below.

Further license information can be found in LICENSE files located in
sub-directories.

Copyright 2014, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

HarfBuzz is licensed under the so-called "Old MIT" license. Details follow.
For parts of HarfBuzz that are licensed under different licenses see individual
files names COPYING in subdirectories where applicable.

Copyright 2019 Facebook, Inc.
Open Source Used In DNAC Platform Arica 5304

Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stribley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
above copyright notice and the following two paragraphs appear in
all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR
DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN
IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS
ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO
PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
(The MIT License)

Copyright (C) 2012 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

qcms
Copyright (C) 2009 Mozilla Corporation
Copyright (C) 1998-2007 Marti Maria

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2017 yhirose

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Public domain
Copyright (C) 2001-2002 by Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*** Portions of the driver (that is, bdflib.c and bdf.h):

Copyright 2000 Computing Research Labs, New Mexico State University
Copyright 2001-2002, 2011 Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

 Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year  name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

⟨signature of Ty Coon⟩, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Copyright 2015 Calvin Metcalf

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

# This is the official list of people who can contribute
# (and typically have contributed) code to the Wuffs repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# The submission process automatically checks to make sure
# that people submitting code are listed in this file (by email address).
#
# Names should be added to this file only after verifying that
# the individual or the individual's organization has agreed to
# the appropriate Contributor License Agreement, found here:
#
#     http://code.google.com/legal/individual-cla-v1.0.html
# The agreement for individuals can be filled out on the web.
#
# When adding J Random Contributor's name to this file, either J's name or J's organization's name should be added to the AUTHORS file, depending on whether the individual or corporate CLA was used.

# Names should be added to this file like so:
# Individual's name <submission email address>
# Individual's name <submission email address> <email2> <emailN>
#
# An entry with multiple email addresses specifies that the first address should be used in the submit logs and that the other addresses should be recognized as the same person when interacting with Gerrit.

# Please keep the list sorted.

Andrew Dassonville <dassonville.andrew@gmail.com>
Jimmy Casey <casey.jimmy@gmail.com>
Nigel Tao <nigeltao@golang.org>
Copyright (c) 2013 Christopher Thunes

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-2018 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
Copyright 2014 The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

# Contributors to the OpenH264 project

Patrick Ai
Sijia Chen
ZhaoZheng Chu
Paley Du
Martin Ettl
Andreas Gal
Xu Guang
Licai Guo
Yi Guo
Horace Huang
Steven Huang
Ethan Hugg
Cullen Jennings
Zhaofeng Jia
Derrick Jin
Jesse Li
Jifei Li
Kai Li
Karina Li
Matt Li
Xiang Li
Bourne Ling
Alex Liu
Wayne Liu
Varun Patil
Eric Rescorla
Adam Roach
Sawyer Shan
Siping Tao
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* MODP_B64 - High performance base64 encoder/decoder
* Version 1.3 -- 17-Mar-2006
* http://modp.com/release/base64
* Copyright (c) 2005, 2006 Nick Galbreath -- nickg [at] modp [dot] com
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are
* met:
* 
* Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 
* Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 
* Neither the name of the modp.com nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
* OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The project in general is under the following licenses:

=================================================================================================
Copyright 2010 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=================================================================================================

File dict_phrases.dat is built from libTabe; the licenses of libTabe is:

=================================================================================================
Copyright (c) 1999 TaBE Project.
Copyright (c) 1999 Pai-Hsiang Hsiao.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
. Neither the name of the TaBE Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999 Computer Systems and Communication Lab,
Institute of Information Science, Academia Sinica.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
. Neither the name of the Computer Systems and Communication Lab
nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 1996 Chih-Hao Tsai @ Beckman Institute, University of Illinois
c-tsai4@uiuc.edu http://casper.beckman.uiuc.edu/~c-tsai4
================================================================================
Copyright (c) 2003-2006, Joe Walnes
Copyright (c) 2006-2015 XStream Committers
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of XStream nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.
Copyright (c) 2016, the R8 project authors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Libevent is available for use under the following license, commonly known
as the 3-clause (or "modified") BSD license:

===============================================
Copyright (c) 2000-2007 Niels Provos <provos@citi.umich.edu>
Copyright (c) 2007-2010 Niels Provos and Nick Mathewson

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================

Portions of Libevent are based on works by others, also made available by them under the three-clause BSD license above. The copyright notices are available in the corresponding source files; the license is as above. Here's a list:

log.c:
Copyright (c) 2000 Dug Song <dugsong@monkey.org>
Copyright (c) 1993 The Regents of the University of California.

strlcpy.c:
Copyright (c) 1998 Todd C. Miller <Todd.Miller@courtesan.com>

win32.c:
Copyright (c) 2003 Michael A. Davis <mike@datanerds.net>

evport.c:
Copyright (c) 2007 Sun Microsystems

min_heap.h:
Copyright (c) 2006 Maxim Yegorushkin <maxim.yegorushkin@gmail.com>

tree.h:
Copyright 2002 Niels Provos <provos@citi.umich.edu>
The MIT License (MIT)

Copyright (c) 2014 Daniel Cousens

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

xxHash Library
Copyright (c) 2012-2014, Yann Collet
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper
Copyright (c) 2001-2017 Expat maintainers

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Blaine Bublitz <blaine.bublitz@gmail.com>, Eric Schoffstall <yo@contra.io> and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

AutobahnAndroid
Copyright 2011,2012 Tavendo GmbH. Licensed under Apache 2.0
This product includes software developed at Tavendo GmbH http://www.tavendo.de
Copyright (c) 2012 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,

Open Source Used In DNAC Platform Arica 5330
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Tencent is pleased to support the open source community by making RapidJSON available.

Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip. All rights reserved.

If you have downloaded a copy of the RapidJSON binary from Tencent, please note that the RapidJSON binary is licensed under the MIT License.

If you have downloaded a copy of the RapidJSON source code from Tencent, please note that RapidJSON source code is licensed under the MIT License, except for the third-party components listed below which are subject to different license terms. Your integration of RapidJSON into your own projects may require compliance with the MIT License, as well as the other licenses applicable to the third-party components included within RapidJSON. To avoid the problematic JSON license in your own projects, it's sufficient to exclude the bin/jsonchecker/ directory, as it's the only code under the JSON license.

A copy of the MIT License is included in this file.

Other dependencies and licenses:

Open Source Software Licensed Under the BSD License:

The msinttypes r29
Copyright (c) 2006-2013 Alexander Chemeris
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Open Source Software Licensed Under the JSON License:
json.org
Copyright (c) 2002 JSON.org
All Rights Reserved.

JSON_checker
Copyright (c) 2002 JSON.org
All Rights Reserved.

Terms of the JSON License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Terms of the MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2014-2015 Devon Govett <devongovett@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This project contains parts of Node.js.
Node.js is licensed for use as follows:

====
Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====
This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

---

Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

---

The Ed25519 software is in the public domain.
Copyright (c) 2011 Alexander Shtuchkin

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2009, The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Software License Agreement (BSD License)

Copyright (c) 2009-2015, Kevin Decker <kpdecker@gmail.com>

All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Kevin Decker nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Javolution - Java(TM) Solution for Real-Time and Embedded Systems
Copyright (c) 2006, Javolution (http://javolution.org)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2017 crypto-browserify contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

W3C 3-clause BSD License

http://www.w3.org/Consortium/Legal/2008/03-bsd-license.html

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of works must retain the original copyright notice,
  this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the original copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

* Neither the name of the W3C nor the names of its contributors may be
  used to endorse or promote products derived from this work without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
The MIT License

Copyright (c) 2011-2015 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2011 Google Inc. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

pyelftools is in the public domain (see below if you need more details).

pyelftools uses the construct library for structured parsing of a binary stream. construct is packaged in pyelftools/construct - see its LICENSE file for the license.

---------------------------------------------------------------------------------------------------------------------------------
This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>

GNU Free Documentation License
Version 1.3, 3 November 2008


Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.
We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may
be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History"). To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other
implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated
location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for
public access to a Transparent copy of the Document, and likewise
the network locations given in the Document for previous versions
it was based on. These may be placed in the "History" section.
You may omit a network location for a work that was published at
least four years before the Document itself, or if the original
publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications",
Preserve the Title of the section, and preserve in the section all
the substance and tone of each of the contributor acknowledgements
and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document,
unaltered in their text and in their titles. Section numbers
or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section
may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled "Endorsements"
or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or
appendices that qualify as Secondary Sections and contain no material
copied from the Document, you may at your option designate some or all
of these sections as invariant. To do this, add their titles to the
list of Invariant Sections in the Modified Version's license notice.
These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains
nothing but endorsements of your Modified Version by various
parties--for example, statements of peer review or that the text has
been approved by an organization as the authoritative definition of a
standard.

You may add a passage of up to five words as a Front-Cover Text, and a
passage of up to 25 words as a Back-Cover Text, to the end of the list
of Cover Texts in the Modified Version. Only one passage of
Front-Cover Text and one of Back-Cover Text may be added by (or
through arrangements made by) any one entity. If the Document already
includes a cover text for the same cover, previously added by you or
by arrangement made by the same entity you are acting on behalf of,
you may not add another; but you may replace the old one, on explicit
permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License
give permission to use their names for publicity for or to assert or
imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS
You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not
apply to the other works in the aggregate which are not themselves
derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these
copies of the Document, then if the Document is less than one half of
the entire aggregate, the Document's Cover Texts may be placed on
covers that bracket the Document within the aggregate, or the
electronic equivalent of covers if the Document is in electronic form.
Otherwise they must appear on printed covers that bracket the whole
aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may
distribute translations of the Document under the terms of section 4.
Replacing Invariant Sections with translations requires special
permission from their copyright holders, but you may include
translations of some or all Invariant Sections in addition to the
original versions of these Invariant Sections. You may include a
translation of this License, and all the license notices in the
Document, and any Warranty Disclaimers, provided that you also include
the original English version of this License and the original versions
of those notices and disclaimers. In case of a disagreement between
the translation and the original version of this License or a notice
or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements",
"Dedications", or "History", the requirement (section 4) to Preserve
its Title (section 1) will typically require changing the actual
title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense, or distribute it is void, and
will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license
from a particular copyright holder is reinstated (a) provisionally,
unless and until the copyright holder explicitly and finally
terminates your license, and (b) permanently, if the copyright holder
fails to notify you of the violation by some reasonable means prior to
60 days after the cessation.

Moreover, your license from a particular copyright holder is
reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.
"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.
Copyright (c) 2010-2014 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

3.1.0 / 2017-09-26
==================
* Add `DEBUG_HIDE_DATE` env var (#486)
* Remove ReDoS regexp in %o formatter (#504)
* Remove `"component"` from package.json
* Remove `component.json`
* Ignore package-lock.json
* Examples: fix colors printout
* Fix: browser detection
* Fix: spelling mistake (#496, @EdwardBetts)

3.0.1 / 2017-08-24
==================
* Fix: Disable colors in Edge and Internet Explorer (#489)

3.0.0 / 2017-08-08
==================
* Breaking: Remove DEBUG_FD (#406)
* Breaking: Use `Date#toISOString()` instead to `Date#toUTCString()` when output is not a TTY (#418)
* Breaking: Make millisecond timer namespace specific and allow 'always enabled' output (#408)
* Addition: document `enabled` flag (#465)
* Addition: add 256 colors mode (#481)
* Addition: `enabled()` updates existing debug instances, add `destroy()` function (#440)
* Update: component: update "ms" to v2.0.0
* Update: separate the Node and Browser tests in Travis-CI
* Update: refactor Readme, fixed documentation, added "Namespace Colors" section, redid screenshots
* Update: separate Node.js and web browser examples for organization
* Update: update "browserify" to v14.4.0
* Fix: fix Readme typo (#473)
2.6.9 / 2017-09-22

* remove ReDoS regexp in %o formatter (#504)

2.6.8 / 2017-05-18

* Fix: Check for undefined on browser globals (#462, @marbemac)

2.6.7 / 2017-05-16

* Fix: Update ms to 2.0.0 to fix regular expression denial of service vulnerability (#458, @hubdotcom)
* Fix: Inline extend function in node implementation (#452, @dougwilson)
* Docs: Fix typo (#455, @msasad)

2.6.5 / 2017-04-27

* Fix: null reference check on window.documentElement.style.WebkitAppearance (#447, @thebigredgeek)
* Misc: clean up browser reference checks (#447, @thebigredgeek)
* Misc: add npm-debug.log to .gitignore (@thebigredgeek)

2.6.4 / 2017-04-20

* Fix: bug that would occur if process.env.DEBUG is a non-string value. (#444, @LucianBuzzo)
* Chore: ignore bower.json in npm installations. (#437, @joaovieira)
* Misc: update "ms" to v0.7.3 (@tootallnate)

2.6.3 / 2017-03-13

* Fix: Electron reference to `process.env.DEBUG` (#431, @paulcbetts)
* Docs: Changelog fix (@thebigredgeek)

2.6.2 / 2017-03-10

* Fix: DEBUG_MAX_ARRAY_LENGTH (#420, @slavaGanzin)
* Docs: Add backers and sponsors from Open Collective (#422, @piamancini)
* Docs: Add Slackin invite badge (@tootallnate)

2.6.1 / 2017-02-10

* Fix: ReDoS regexp in %o formatter (#504)
* Fix: Module's `export default` syntax fix for IE8 `Expected identifier` error
* Fix: Whitelist DEBUG_FD for values 1 and 2 only (#415, @pi0)
* Fix: IE8 "Expected identifier" error (#414, @vgoma)
* Fix: Namespaces would not disable once enabled (#409, @musikov)

2.6.0 / 2016-12-28
==================

* Fix: added better null pointer checks for browser useColors (@thebigredgeek)
* Improvement: removed explicit `window.debug` export (#404, @tootallnate)
* Improvement: deprecated `DEBUG_FD` environment variable (#405, @tootallnate)

2.5.2 / 2016-12-25
==================

* Fix: reference error on window within webworkers (#393, @KlausTrainer)
* Docs: fixed README typo (#391, @lurch)
* Docs: added notice about v3 api discussion (@thebigredgeek)

2.5.1 / 2016-12-20
==================

* Fix: babel-core compatibility

2.5.0 / 2016-12-20
==================

* Fix: wrong reference in bower file (@thebigredgeek)
* Fix: webworker compatibility (@thebigredgeek)
* Fix: output formatting issue (#388, @kribblo)
* Fix: babel-loader compatibility (#383, @escwald)
* Misc: removed built asset from repo and publications (@thebigredgeek)
* Misc: moved source files to /src (#378, @yamikuronue)
* Test: added karma integration and replaced babel with browserify for browser tests (#378, @yamikuronue)
* Test: coveralls integration (#378, @yamikuronue)
* Docs: simplified language in the opening paragraph (#373, @yamikuronue)

2.4.5 / 2016-12-17
==================

* Fix: `navigator` undefined in Rhino (#376, @jochenberger)
* Fix: custom log function (#379, @hsiliev)
* Improvement: bit of cleanup + linting fixes (@thebigredgeek)
* Improvement: rm non-maintained `dist/` dir (#375, @freewil)
* Docs: simplified language in the opening paragraph. (#373, @yamikuronue)

2.4.4 / 2016-12-14
==================
* Fix: work around debug being loaded in preload scripts for electron (#368, @paulcbetts)

2.4.3 / 2016-12-14
==================

* Fix: navigation.userAgent error for react native (#364, @escwald)

2.4.2 / 2016-12-14
==================

* Fix: browser colors (#367, @tootallnate)
* Misc: travis ci integration (@thebigredgeek)
* Misc: added linting and testing boilerplate with sanity check (@thebigredgeek)

2.4.1 / 2016-12-13
==================

* Fix: typo that broke the package (#356)

2.4.0 / 2016-12-13
==================

* Fix: bower.json references unbuilt src entry point (#342, @justmatt)
* Fix: revert "handle regex special characters" (@tootallnate)
* Feature: configurable util.inspect()’options for NodeJS (#327, @tootallnate)
* Feature: %O (big O) pretty-prints objects (#322, @tootallnate)
* Improvement: allow colors in workers (#335, @botverse)
* Improvement: use same color for same namespace. (#338, @lchenay)

2.3.3 / 2016-11-09
==================

* Fix: Catch `JSON.stringify()` errors (#195, Jovan Alleyne)
* Fix: Returning `localStorage` saved values (#331, Levi Thomason)
* Improvement: Don't create an empty object when no `process` (Nathan Rajlich)

2.3.2 / 2016-11-09
==================

* Fix: be super-safe in index.js as well (@TooTallNate)
* Fix: should check whether process exists (Tom Newby)

2.3.1 / 2016-11-09
==================

* Fix: Added electron compatibility (#324, @paulcbetts)
* Improvement: Added performance optimizations (@tootallnate)
* Readme: Corrected PowerShell environment variable example (#252, @gimre)
* Misc: Removed yarn lock file from source control (#321, @fengmk2)

2.3.0 / 2016-11-07
==================

* Fix: Consistent placement of ms diff at end of output (#215, @gorangajic)
* Fix: Escaping of regex special characters in namespace strings (#250, @zacronos)
* Fix: Fixed bug causing crash on react-native (#282, @vkarpov15)
* Feature: Enabled ES6+ compatible import via default export (#212 @bucaran)
* Feature: Added `%O` formatter to reflect Chrome's console.log capability (#279, @oncletom)
* Package: Update "ms" to 0.7.2 (#315, @DevSide)
* Package: removed superfluous version property from bower.json (#207 @kkirsche)
* Readme: fix USE.COLORS to DEBUG.COLORS
* Readme: Doc fixes for format string sugar (#269, @mlucool)
* Readme: Updated docs for DEBUG_FD and DEBUG.COLORS environment variables (#232, @mattlyons0)
* Readme: doc fixes for PowerShell (#271 #243, @exoticknight @unreadable)
* Readme: better docs for browser support (#224, @matthewmueller)
* Tooling: Added yarn integration for development (#317, @thebigredgeek)
* Misc: Renamed History.md to CHANGELOG.md (@thebigredgeek)
* Misc: Added license file (#226 #274, @CantemoInternal @sdaitzman)
* Misc: Updated contributors (@thebigredgeek)

2.2.0 / 2015-05-09
==================

* package: update "ms" to v0.7.1 (#202, @dougwilson)
* README: add logging to file example (#193, @DanielOchoa)
* README: fixed a typo (#191, @amir-s)
* browser: expose `storage` (#190, @stephenmathieson)
* Makefile: add a `distclean` target (#189, @stephenmathieson)

2.1.3 / 2015-03-13
==================

* Updated stdout/stderr example (#186)
* Updated example/stdout.js to match debug current behaviour
* Renamed example/stderr.js to stdout.js
* Update Readme.md (#184)
* replace high intensity foreground color for bold (#182, #183)

2.1.2 / 2015-03-01
==================

* dist: recompile
* update "ms" to v0.7.0
* package: update "browserify" to v9.0.3
* component: fix "ms.js" repo location
* changed bower package name
* updated documentation about using debug in a browser
* fix: security error on safari (#167, #168, @yields)

2.1.1 / 2014-12-29
==================

* browser: use `typeof` to check for `console` existence
* browser: check for `console.log` truthiness (fix IE 8/9)
* browser: add support for Chrome apps
* Readme: added Windows usage remarks
  * Add `bower.json` to properly support bower install

2.1.0 / 2014-10-15
==================

* node: implement `DEBUG_FD` env variable support
* package: update "browserify" to v6.1.0
* package: add "license" field to package.json (#135, @panuhorsmalaihti)

2.0.0 / 2014-09-01
==================

* package: update "browserify" to v5.11.0
* node: use stderr rather than stdout for logging (#29, @stephenmathieson)

1.0.4 / 2014-07-15
==================

* dist: recompile
* example: remove `console.info()` log usage
* example: add "Content-Type" UTF-8 header to browser example
* browser: place `%c` marker after the space character
* browser: reset the "content" color via `color: inherit`
* browser: add colors support for Firefox >= v31
* debug: prefer an instance `log()` function over the global one (#119)
  * Readme: update documentation about styled console logs for FF v31 (#116, @wryk)

1.0.3 / 2014-07-09
==================

* Add support for multiple wildcards in namespaces (#122, @seegno)
* browser: fix lint

1.0.2 / 2014-06-10
==================

* browser: update color palette (#113, @gscottolson)
* common: make console logging function configurable (#108, @timoxley)
* node: fix `%o` colors on old node <= 0.8.x
* Makefile: find node path using shell/which (#109, @timoxley)

1.0.1 / 2014-06-06
==================

* browser: use `removItem()` to clear localStorage
* browser, node: don't set DEBUG if namespaces is undefined (#107, @leedm777)
* package: add "contributors" section
* node: fix comment typo
* README: list authors

1.0.0 / 2014-06-04
=================

* make ms diff be global, not be scope
* debug: ignore empty strings in enable()
* node: make DEBUG_COLORS able to disable coloring
* *: export the `colors` array
* npmignore: don't publish the `dist` dir
* Makefile: refactor to use browserify
* package: add "browserify" as a dev dependency
* Readme: add Web Inspector Colors section
* node: reset terminal color for the debug content
* node: map `%o` to `util.inspect()`
* browser: map `%j` to `JSON.stringify()`
* debug: add custom "formatters"
* debug: use "ms" module for humanizing the diff
* Readme: add "bash" syntax highlighting
* browser: add Firebug color support
* browser: add colors for WebKit browsers
* node: apply log to `console`
* rewrite: abstract common logic for Node & browsers
* add .jshintrc file

0.8.1 / 2014-04-14
=================

* package: re-add the "component" section

0.8.0 / 2014-03-30
=================

* add `enable()` method for nodejs. Closes #27
* change from stderr to stdout
* remove unnecessary index.js file
0.7.4 / 2013-11-13

* remove "browserify" key from package.json (fixes something in browserify)

0.7.3 / 2013-10-30

* fix: catch localStorage security error when cookies are blocked (Chrome)
* add debug(err) support. Closes #46
* add .browser prop to package.json. Closes #42

0.7.2 / 2013-02-06

* fix package.json
* fix: Mobile Safari (private mode) is broken with debug
* fix: Use unicode to send escape character to shell instead of octal to work with strict mode javascript

0.7.1 / 2013-02-05

* fix component. Closes #24

0.7.0 / 2012-05-04

* Added .component to package.json
* Added debug.component.js build

0.6.0 / 2012-03-16

* Added support for "." prefix in DEBUG [Vinay Pulim]
* Added `enabled` flag to the node version [TooTallNate]

0.5.0 / 2012-02-02

* Added: humanize diffs. Closes #8
* Added `debug.disable()` to the CS variant
* Removed padding. Closes #10
* Fixed: persist client-side variant again. Closes #9

0.4.0 / 2012-02-01
* Added browser variant support for older browsers [TooTallNate]
* Added `debug.enable('project:*')` to browser variant [TooTallNate]
* Added padding to diff (moved it to the right)

0.3.0 / 2012-01-26

* Added millisecond diff when isatty, otherwise UTC string

0.2.0 / 2012-01-22

* Added wildcard support

0.1.0 / 2011-12-02

* Added: remove colors unless stderr isatty [TooTallNate]

0.0.1 / 2010-01-03

* Initial release

MIT License

Copyright (c) 2012-2019 Thorsten Lorenz, Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright (c) 2005-2008, The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
// Copyright 2014 The Chromium Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:

// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)
Copyright 1991-2017 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in http://www.unicode.org/copyright.html

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.
THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional terms for licensed third-party software components included within ICU libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others  All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE SOFTWARE.
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.
Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

For the file sRGB2014.icc:

Copyright International Color Consortium, 2015

This profile is made available by the International Color Consortium, and may be copied, distributed, embedded, made, used, and sold without restriction. Altered versions of this profile shall have the original identification and copyright information removed and shall not be misrepresented as the original profile.

(original source location: http://www.color.org/srgbprofiles.xalter)
Copyright 2007, 2008 The Python Markdown Project (v. 1.7 and later)
Copyright 2004, 2005, 2006 Yuri Takhteyev (v. 0.2-1.6b)
Copyright 2004 Manfred Stienstra (the original version)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE PYTHON MARKDOWN PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ANY CONTRIBUTORS TO THE PYTHON MARKDOWN PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) Sam Verschueren <sam.verschueren@gmail.com> (github.com/SamVerschueren)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

musl as a whole is licensed under the following standard MIT license:

------------------------------------------------------------------------

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

----------------------------------------------------------------------

Authors/contributors include:

Alex Dowad
Alexander Monakov
Anthony G. Basile
Arvid Picciani
Bobby Bingham
Boris Brezillon
Brent Cook
Chris Spiegel
Clment Vasseur
Daniel Micay
Denys Vlasenko
Emil Renner Berthing
Felix Fietkau
Felix Janda
Gianluca Anzolin
Hauke Mehrtens
Hiltjo Posthuma
Isaac Dunham
Jaydeep Patil
Jens Gustedt
Jeremy Huntwork
Jo-Philipp Wich
Joakim Sindholt
John Spencer
Josiah Worcester
Justin Cormack
Khem Raj
Kylie McClain
Luca Barbato
Luka Perkov
M Farkas-Dyck (Strake)
Mahesh Bodapati
Michael Forney
Natanael Copa
Nicholas J. Kain
orc
Pascal Cuq
Petr Hosek
Pierre Carrier
 Portions of this software are derived from third-party works licensed under terms compatible with the above MIT license:

The TRE regular expression implementation (src/regex/reg* and src/regex/tre*) is Copyright 2001-2008 Ville Laurikari and licensed under a 2-clause BSD license (license text in the source files). The included version has been heavily modified by Rich Felker in 2012, in the interests of size, simplicity, and namespace cleanliness.

Much of the math library code (src/math/* and src/complex/*) is Copyright 1993,2004 Sun Microsystems or Copyright 2003-2011 David Schultz or Copyright 2003-2009 Steven G. Kargl or Copyright 2003-2009 Bruce D. Evans or Copyright 2008 Stephen L. Moshier and labelled as such in comments in the individual source files. All have been licensed under extremely permissive terms.

The ARM memcpy code (src/string/arm/memcpy_el.S) is Copyright 2008 The Android Open Source Project and is licensed under a two-clause BSD license. It was taken from Bionic libc, used on Android.

The implementation of DES for crypt (src/crypt/crypt_des.c) is Copyright 1994 David Burren. It is licensed under a BSD license.

The implementation of blowfish crypt (src/crypt/crypt_blowfish.c) was originally written by Solar Designer and placed into the public domain. The code also comes with a fallback permissive license for use in jurisdictions that may not recognize the public domain.

The smoothsort implementation (src/stdlib/qsort.c) is Copyright 2011 Valentin Ochs and is licensed under an MIT-style license.

The BSD PRNG implementation (src/prng/random.c) and XSI search API (src/search/*.c) functions are Copyright 2011 Szabolcs Nagy and licensed under following terms: "Permission to use, copy, modify,
and/or distribute this code for any purpose with or without fee is hereby granted. There is no warranty.”

The x86_64 port was written by Nicholas J. Kain and is licensed under the standard MIT terms.

The mips and microblaze ports were originally written by Richard Pennington for use in the ellcc project. The original code was adapted by Rich Felker for build system and code conventions during upstream integration. It is licensed under the standard MIT terms.

The mips64 port was contributed by Imagination Technologies and is licensed under the standard MIT terms.

The powerpc port was also originally written by Richard Pennington, and later supplemented and integrated by John Spencer. It is licensed under the standard MIT terms.

All other files which have no copyright comments are original works produced specifically for use as part of this library, written either by Rich Felker, the main author of the library, or by one or more contributors listed above. Details on authorship of individual files can be found in the git version control history of the project. The omission of copyright and license comments in each file is in the interest of source tree size.

In addition, permission is hereby granted for all public header files (include/* and arch/*/bits/*) and crt files intended to be linked into applications (crt/*, ldso/dlstart.c, and arch/*/crt_arch.h) to omit the copyright notice and permission notice otherwise required by the license, and to use these files without any requirement of attribution. These files include substantial contributions from:

Bobby Bingham
John Spencer
Nicholas J. Kain
Rich Felker
Richard Pennington
Stefan Kristiansson
Szabolcs Nagy

all of whom have explicitly granted such permission.

This file previously contained text expressing a belief that most of the files covered by the above exception were sufficiently trivial not to be subject to copyright, resulting in confusion over whether it negated the permissions granted in the license. In the spirit of permissive licensing, and of not having licensing issues being an
obstacle to adoption, that text has been removed.
Copyright (c) 1994-2006 Sun Microsystems Inc.
All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

- Redistribution in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

- Neither the name of Sun Microsystems or the names of contributors may
  be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License

Copyright (c) 2015 Kyle E. Mitchell & other authors listed in AUTHORS

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(WebKit doesn't distribute an explicit license. This LICENSE is derived from license text in the source.)


The terms and conditions vary from file to file, but are one of:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*OR*

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Apple Computer, Inc. ("Apple") nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY APPLE COMPUTER, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE COMPUTER, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that
they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or
table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot

---

**Open Source Used In DNAC Platform Arica 5392**
distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The
former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or
9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS
Copyright (C) 2012-2018 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
(c) 2007-2009 Steven Levithan <stevenlevithan.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

BSD 3-Clause License

Copyright (c) 2005 - 2017, Jochen Kalmbach
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright 2015 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2010-2018 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)

Copyright (c) Node.js contributors, James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2013 Rase-

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2010, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Additional IP Rights Grant (Patents)
------------------------------------

"These implementations" means the copyrightable works that implement the WebM codecs distributed by Google as part of the WebM Project.

Google hereby grants to you a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, transfer, and otherwise run, modify and propagate the contents of these implementations of WebM, where such license applies only to those patent claims, both currently owned by Google and acquired in the future, licensable by Google that are necessarily infringed by these implementations of WebM. This grant does not include claims that would be infringed only as a consequence of further modification of these implementations. If you or your agent or exclusive licensee institute or order or agree to the institution of patent litigation or any other patent enforcement activity against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that any of these implementations of WebM or any code incorporated within any of these implementations of WebM constitute direct or contributory patent infringement, or inducement of patent infringement, then any patent rights granted to you under this License for these implementations of WebM shall terminate as of the date such litigation is filed.

The MIT License (MIT)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# License

Most files in FFmpeg are under the GNU Lesser General Public License version 2.1 or later (LGPL v2.1+). Read the file `COPYING.LGPLv2.1` for details. Some other files have MIT/X11/BSD-style licenses. In combination the LGPL v2.1+ applies to FFmpeg.

Some optional parts of FFmpeg are licensed under the GNU General Public License version 2 or later (GPL v2+). See the file `COPYING.GPLv2` for details. None of these parts are used by default, you have to explicitly pass `--enable-gpl` to configure to activate them. In this case, FFmpeg's license changes to GPL v2+.

Specifically, the GPL parts of FFmpeg are:

- libpostproc
- optional x86 optimization in the files
  - `libavcodec/x86/flac_dsp_gpl.asm`
  - `libavcodec/x86/idct_mmx.c`
  - `libavfilter/x86/vf_removegrain.asm`
- the following building and testing tools
  - `compat/solaris/make_sunver.pl`
  - `doc/t2h.pm`
  - `doc/texi2pod.pl`
  - `libswresample/tests/swresample.c`
  - `tests/checkasm/*`
  - `tests/tiny_ssim.c`
- the following filters in libavfilter:
  - `signature_lookup.c`
- `vf_blackframe.c`
- `vf_boxblur.c`
- `vf_colormatrix.c`
- `vf_cover_rect.c`
- `vf_cropdetect.c`
- `vf_delogo.c`
- `vf_eq.c`
- `vf_find_rect.c`
- `vf_fspp.c`
- `vf_geq.c`
- `vf_histeq.c`
- `vf_hqdn3d.c`
- `vf_kerndeint.c`
- `vf_lensfun.c` (GPL version 3 or later)
- `vf_mcdeint.c`
- `vf_mpdecimate.c`
- `vf_nnedi.c`
- `vf_owdenoise.c`
- `vf_perspective.c`
- `vf_phase.c`
- `vf_pp.c`
- `vf_pp7.c`
- `vf_pullup.c`
- `vf_repeatfields.c`
- `vf_sab.c`
- `vf_signature.c`
- `vf_smartblur.c`
- `vf_spp.c`
- `vf_stereo3d.c`
- `vf_super2xsai.c`
- `vf_tinterlace.c`
- `vf_uspp.c`
- `vf_vaguedenoiser.c`
- `vsrc_mptestsrc.c`

Should you, for whatever reason, prefer to use version 3 of the (L)GPL, then the configure parameter `--enable-version3` will activate this licensing option for you. Read the file `COPYING.LGPLv3` or, if you have enabled GPL parts, `COPYING.GPLv3` to learn the exact legal terms that apply in this case.

There are a handful of files under other licensing terms, namely:

* The files `libavcodec/jfdctfst.c`, `libavcodec/jfdctint_template.c` and `libavcodec/jrevdct.c` are taken from libjpeg, see the top of the files for licensing details. Specifically note that you must credit the IJG in the documentation accompanying your program if you only distribute executables. You must also indicate any changes including additions and deletions to those three files in the documentation.
## External libraries

FFmpeg can be combined with a number of external libraries, which sometimes affect the licensing of binaries resulting from the combination.

### Compatible libraries

The following libraries are under GPL version 2:
- avisynth
- frei0r
- libcdio
- libdavs2
- librubberband
- libvidstab
- libx264
- libx265
- libxavs
- libxavs2
- libxvid

When combining them with FFmpeg, FFmpeg needs to be licensed as GPL as well by passing `--enable-gpl` to configure.

The following libraries are under LGPL version 3:
- gmp
- libaribb24
- liblensfun

When combining them with FFmpeg, use the configure option `--enable-version3` to upgrade FFmpeg to the LGPL v3.

The VMAF, mbedTLS, RK MPI, OpenCORE and VisualOn libraries are under the Apache License 2.0. That license is incompatible with the LGPL v2.1 and the GPL v2, but not with version 3 of those licenses. So to combine these libraries with FFmpeg, the license version needs to be upgraded by passing `--enable-version3` to configure.

The smbclient library is under the GPL v3, to combine it with FFmpeg, the options `--enable-gpl` and `--enable-version3` have to be passed to configure to upgrade FFmpeg to the GPL v3.

### Incompatible libraries

There are certain libraries you can combine with FFmpeg whose licenses are not compatible with the GPL and/or the LGPL. If you wish to enable these libraries, even in circumstances that their license may be incompatible, pass
'--enable-nonfree' to configure. This will cause the resulting binary to be unredistributable.

The Fraunhofer FDK AAC and OpenSSL libraries are under licenses which are incompatible with the GPLv2 and v3. To the best of our knowledge, they are compatible with the LGPL.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2014 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2017 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd. Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below. This license makes reference to the version 3 of the GNU General Public License, which you can find in the LICENSE.GPLv3 file.

-------------------------------------------------------------------------

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this
licensedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.
0. Additional Definitions.

As used herein, this License refers to version 3 of the GNU Lesser General Public License, and the GNU GPL refers to version 3 of the GNU General Public License.

The Library refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An Application is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A Combined Work is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the Linked Version.

The Minimal Corresponding Source for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The Corresponding Application Code for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.
1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run
time a copy of the Library already present on the user's
computer system, and (b) will operate properly with a modified
version of the Library that is interface-compatible with the
Linked Version.

e) Provide Installation Information, but only if you would
otherwise be required to provide such information under section 6
of the GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the
Combined Work produced by recombining or relinking the Application
with a modified version of the Linked Version. (If you use option
4d0, the Installation Information must accompany the Minimal
Corresponding Source and Corresponding Application Code. If you
use option 4d1, you must provide the Installation Information in
the manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.)


You may place library facilities that are a work based on the Library
side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

   a) Accompany the combined library with a copy of the same work
      based on the Library, uncombined with any other library
      facilities, conveyed under the terms of this License.

   b) Give prominent notice with the combined library that part of
      it is a work based on the Library, and explaining where to find
      the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
as you received it specifies that a certain numbered version of the
GNU Lesser General Public License or any later version applies to
it, you have the option of following the terms and conditions either
of that published version or of any later version published by the
Free Software Foundation. If the Library as you received it does not
specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

******    *****   ******   UnRAR - free utility for RAR archives
**   **  **   **  **   **  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
******   *******  ******    License for use and distribution of
**   **  **   **  **   **   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
**   **  **   **  **   **         FREE portable version
~~~~~~~~~~~~~~~~~~~~~

The source code of UnRAR utility is freeware. This means:

1. All copyrights to RAR and the utility UnRAR are exclusively owned by the author - Alexander Roshal.

2. UnRAR source code may be used in any software to handle RAR archives without limitations free of charge, but cannot be used to develop RAR (WinRAR) compatible archiver and to re-create RAR compression algorithm, which is proprietary. Distribution of modified UnRAR source code in separate form or as a part of other software is permitted, provided that full text of this paragraph, starting from "UnRAR source code" words, is included in license, or in documentation if license is not available, and in source code comments of resulting package.

3. The UnRAR utility may be freely distributed. It is allowed to distribute UnRAR inside of other software packages.

4. THE RAR ARCHIVER AND THE UnRAR UTILITY ARE DISTRIBUTED "AS IS". NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED. YOU USE AT YOUR OWN RISK. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS, DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING OR MISUSING THIS SOFTWARE.

5. Installing and using the UnRAR utility signifies acceptance of these terms and conditions of the license.

6. If you don't agree with terms of the license you must remove UnRAR files from your storage devices and cease to use the utility.

Thank you for your interest in RAR and UnRAR.
Alexander L. Roshal
S2 Geometry Library
Copyright 2017 Google Inc. All Rights Reserved.

This product includes software developed at Google (https://www.google.com/).
# The 3-Clause BSD License

Copyright 2019 web-platform-tests contributors

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2018 Dmitry Shirokov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright 2010-2017 Mike Bostock
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the author nor the names of contributors may be used to
  endorse or promote products derived from this software without specific prior
  written permission.

THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge,
to any person obtaining a copy of this software and
associated documentation files (the "Software"), to
deal in the Software without restriction, including
without limitation the rights to use, copy, modify,
merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom
the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice
shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/* ***** BEGIN LICENSE BLOCK *****
 * Version: MPL 1.1/GPL 2.0/LGPL 2.1
 * 
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.mozilla.org/MPL/
 * 
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 * 
 * The Original Code is the Netscape Portable Runtime (NSPR).
 * 
 * The Initial Developer of the Original Code is
 * Netscape Communications Corporation.
 * Portions created by the Initial Developer are Copyright (C) 1998-2000
 * the Initial Developer. All Rights Reserved.
 * 
 * Contributor(s):
 * 
 * Alternatively, the contents of this file may be used under the terms of
 * either the GNU General Public License Version 2 or later (the "GPL"), or
 * the GNU Lesser General Public License Version 2.1 or later (the "LGPL"),
 * in which case the provisions of the GPL or the LGPL are applicable instead
 * of those above. If you wish to allow use of your version of this file only
 * under the terms of either the GPL or the LGPL, and not to allow others to
 * use your version of this file under the terms of the MPL, indicate your
 * decision by deleting the provisions above and replace them with the notice
 * and other provisions required by the GPL or the LGPL. If you do not delete
 * the provisions above, a recipient may use your version of this file under
 * the terms of any one of the MPL, the GPL or the LGPL.
 * 
 * ***** END LICENSE BLOCK ***** */

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2014 International Business Machines Corporation and others

All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Third-Party Software Licenses

This section contains third-party software notices and/or additional terms for licensed third-party software components included within ICU libraries.

1. Unicode Data Files and Software

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2014 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that
(a) this copyright and permission notice appear with all copies of the Data Files or Software,
(b) this copyright and permission notice appear in associated documentation, and
(c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.


# The Google Chrome software developed by Google is licensed under the BSD license. Other software included in this distribution is provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
#
# Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

# Neither the name of Google Inc. nor the names of its contributors may be
used to endorse or promote products derived from this software without specific
prior written permission.

#

# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS I
S" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPL
IED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLA IMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIR
ECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDIN
G, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF L
IABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED O
F THE POSSIBILITY OF SUCH DAMAGE.

#

# The word list in cjdict.txt are generated by combining three word lists l
isted
# below with further processing for compound word breaking. The frequency i
s generated
# with an iterative training against Google web corpora.

# *

# Lihtabe (Chinese)
# - https://sourceforge.net/project/?group_id=1519
# - Its license terms and conditions are shown below.

# *

# IPADIC (Japanese)
# - http://chasen.aist-nara.ac.jp/chasen/distribution.html
# - Its license terms and conditions are shown below.

# **********COPYING.libtabe ----------- BEGIN-------------------

# /*

# Copyright (c) 1999 TaBE Project.
# Copyright (c) 1999 Pai-Hsiang Hsiao.
# All rights reserved.

# * Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:

# * Redistributions of source code must retain the above copyright
# notice, this list of conditions and the following disclaimer.

# * Redistributions in binary form must reproduce the above copyright
# notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * . Neither the name of the TaBE Project nor the names of its
# * contributors may be used to endorse or promote products derived
# * from this software without specific prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY. WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# /*
# * Copyright (c) 1999 Computer Systems and Communication Lab,
# * Institute of Information Science, Academia Sinica.
# * All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * . Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * . Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * . Neither the name of the Computer Systems and Communication Lab
# * nor the names of its contributors may be used to endorse or
# * promote products derived from this software without specific
# * prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute, University of Illinois
# c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
#
# ------------COPYING.libtabe-----END-------------------------------
#
#
# ------------COPYING.ipadic-----BEGIN-------------------------------
--
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology.  All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software.  The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to have agreed and understood, that there is no warranty whatsoever for the program and, accordingly, the entire risk arising from or otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other organization that participated in or was otherwise related to the development of the program and their respective officials, directors, officers and other employees shall be held liable for any and all damages, including, without limitation, general, special, incidental and consequential damages, arising out of or otherwise in connection with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

3. Lao Word Break Dictionary Data (laodict.txt)

Copyright (c) 2013 International Business Machines Corporation and others. All Rights Reserved.

Project: http://code.google.com/p/lao-dictionary/
4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

7. Database Ownership

The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it.

Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

Copyright (C) 1993-2004 by Sun Microsystems, Inc. All rights reserved.

Developed at SunSoft, a Sun Microsystems, Inc. business.
Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

GNU GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3. That license references the General Public License version 3, that is displayed below. Other portions of the Qt Toolkit may be licensed directly under this license.

-------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.
2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to
produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified
it, and giving a relevant date.

b) The work must carry prominent notices stating that it is
released under this License and any conditions added under section
7. This requirement modifies the requirement in section 4 to
"keep intact all notices".

c) You must license the entire work, as a whole, under this
License to anyone who comes into possession of a copy. This
License will therefore apply, along with any applicable section 7
additional terms, to the whole of the work, and all its parts,
regardless of how they are packaged. This License gives no
permission to license the work in any other way, but it does not
invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display
Appropriate Legal Notices; however, if the Program has interactive
interfaces that do not display Appropriate Legal Notices, your
work need not make them do so.

A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
“aggregate” if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or
modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and
propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

NAME: WebDriver
URL: http://selenium.googlecode.com/svn/trunk/py
LICENSE: Apache 2
The MIT License (MIT)

Copyright (c) 2013-2015 Roman Shtylman <shtylman@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012-2014 Roman Shtylman <shtylman@gmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**************************************************************************
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
**
** This file is part of the FOO module of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:GPL$

---

Open Source Used In DNAC Platform Arica 5451
Copyright (c) 2013 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2017 Made With MOXY Lda <hello@moxy.studio>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/* ***** BEGIN LICENSE BLOCK *****
* Version: MPL 1.1/GPL 2.0/LGPL 2.1
*
* The contents of this file are subject to the Mozilla Public License Version
* 1.1 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* http://www.mozilla.org/MPL/
*
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*
* The Original Code is mozilla.org code.
*
* The Initial Developer of the Original Code is
* Netscape Communications Corporation.
* Portions created by the Initial Developer are Copyright (C) 2001
* the Initial Developer. All Rights Reserved.
*
* Contributor(s):
*
* Alternatively, the contents of this file may be used under the terms of
* either the GNU General Public License Version 2 or later (the "GPL"), or
* the GNU Lesser General Public License Version 2.1 or later (the "LGPL"),
* in which case the provisions of the GPL or the LGPL are applicable instead
* of those above. If you wish to allow use of your version of this file only
* under the terms of either the GPL or the LGPL, and not to allow others to
* use your version of this file under the terms of the MPL, indicate your
* decision by deleting the provisions above and replace them with the notice
* and other provisions required by the GPL or the LGPL. If you do not delete
* the provisions above, a recipient may use your version of this file under
* the terms of any one of the MPL, the GPL or the LGPL.
*
* ***** END LICENSE BLOCK ***** */
Copyright (c) 2015-2017 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

The MIT License (MIT)

Copyright (c) 2014 Josh Wolfe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2003-2012, Michael Foord
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2017 Chai.js Assertion Library

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2003,2004 Colin Percival
All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted providing that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

//*********************************************************************
//* Base64 - a simple base64 encoder and decoder.
//*
//*     Copyright (c) 1999, Bob Withers - bwit@pobox.com
//*     This code may be freely used for any purpose, either personal
//*     or commercial, provided the authors copyright notice remains
//*     intact.
//*     Enhancements by Stanley Yamane:
//*     o reverse lookup table for the decode function
//*     o reserve string buffer space in advance
//* //*********************************************************************

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.
The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>  

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. 

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details. 

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA 

Also add information on how to contact you by electronic and paper mail. 

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names: 

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker. 

<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice 

That's all there is to it! 

wcwidth.js: JavaScript Portng of Markus Kuhn's wcwidth() Implementation 
======================================================================= 

Copyright (C) 2012 by Jun Woong. 

This package is a JavaScript porting of `wcwidth()` implementation [by Markus Kuhn][http://www.cl.cam.ac.uk/~mgk25/ucs/wcwidth.c]. 

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: 

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2009-2017 The OTS Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2013 Gareth Jones

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

LICENSE
=======
This is an installer created using CPack (http://www.cmake.org). No license provided.
// Copyright (c) 2006-2009 The Chromium OS Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
//   copyright notice, this list of conditions and the following disclaimer
//   in the documentation and/or other materials provided with the
//   distribution.
// * Neither the name of Google Inc. nor the names of its
//   contributors may be used to endorse or promote products derived from
//   this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)
Copyright (c) 2012-2014 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015, Rebecca Turner <me@re-becca.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 2006-2019 the Mako authors and contributors <see AUTHORS file>.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# timers-browserify

This project uses the [MIT](http://jryans.mit-license.org/) license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# lib/node

The `lib/node` directory borrows files from joyent/node which uses the following license:

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

www.solarsystemscope.com/textures distributed under the terms of https://creativecommons.org/licenses/by/4.0/

Changes:
- Rings created from the reference rectangle file.
NSS is available under the Mozilla Public License, version 2, a copy of which is below.

Note on GPL Compatibility
----------------------------

The MPL 2, section 3.3, permits you to combine NSS with code under the GNU General Public License (GPL) version 2, or any later version of that license, to make a Larger Work, and distribute the result under the GPL. The only condition is that you must also make NSS, and any changes you have made to it, available to recipients under the terms of the MPL 2 also.

Anyone who receives the combined code from you does not have to continue to dual licence in this way, and may, if they wish, distribute under the terms of either of the two licences - either the MPL alone or the GPL alone. However, we discourage people from distributing copies of NSS under the GPL alone, because it means that any improvements they make cannot be reincorporated into the main version of NSS. There is never a need to do this for license compatibility reasons.

Note on LGPL Compatibility
-----------------------------

The above also applies to combining MPLed code in a single library with code under the GNU Lesser General Public License (LGPL) version 2.1, or any later version of that license. If the LGPLed code and the MPLed code are not in the same library, then the copyleft coverage of the two licences does not overlap, so no issues arise.

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
    means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.
1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
    in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
    version 1.1 or earlier of the License, but not also under the
    terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
    Software; or

(b) any new file in Source Code Form that contains any Covered
    Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.
1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
    Licensable by such Contributor to use, reproduce, make available,
    modify, display, perform, distribute, and otherwise exploit its
    Contributions, either on an unmodified basis, with Modifications, or
    as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
    for sale, have made, import, and otherwise transfer either its
    Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software;

or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form
All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered
Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                    *
*  6. Disclaimer of Warranty                                        *
*  -------------------------                                        *
*                                                                    *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.
*                                                                    *
************************************************************************
************************************************************************
*                                                                    *
*  7. Limitation of Liability                                      *
*  --------------------------                                      *
*                                                                    *
*  Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.
*                                                                    *
************************************************************************

8. Litigation
----------
Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
of the License under which You originally received the Covered Software,
or under the terms of any subsequent version published by the license
steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a
modified version of this License if you rename the license and remove
any references to the name of the license steward (except to note that
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary
Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the
notice described in Exhibit B of this License must be attached.
Exhibit A - Source Code Form License Notice
-------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
Copyright (c) 2017-2018 Advanced Micro Devices, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.

PEXPECT LICENSE

This license is approved by the OSI and FSF as GPL-compatible.
Copyright (c) 2012, Noah Spurrier <noah@noah.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. The software is provided "as is" and the author disclaims all warranties with regard to this software including all implied warranties of merchantability and fitness. In no event shall the author be liable for any special, direct, indirect, or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Copyright (C) 2005  Michael Ahlberg, Mns Rullgrd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in the software.

Use, modification & distribution is subject to Boost Software License Ver 1.

http://www.boost.org/LICENSE_1_0.txt

Attributions:
The code in this library is an extension of Bala Vatti's clipping algorithm:
"A generic solution to polygon clipping"
http://portal.acm.org/citation.cfm?id=129906

Computer graphics and geometric modeling: implementation and algorithms
By Max K. Agoston
Springer; 1 edition (January 4, 2005)
http://books.google.com/books?q=vatti+clipping+agoston
See also:
"Polygon Offsetting by Computing Winding Numbers"
Paper no. DETC2005-85513 pp. 565-575
ASME 2005 International Design Engineering Technical Conferences
and Computers and Information in Engineering Conference (IDETC/CIE2005)
September 24-28, 2005, Long Beach, California, USA
http://www.me.berkeley.edu/~mcmains/pubs/DAC05OffsetPolygon.pdf
Licenses of dependencies as produced by `cargo-license`:

Apache-2.0/MIT (20): js-sys, auto cfg, cc, indexmap, wasm-bindgen, wasm-bindgen-macro-support, lazy_static,
gimli, fallible-iterator, arrayvec, wasm-bindgen-macro, wasm-bindgen-shared, backtrace-sys, backtrace, wasm-
bindgen-backend, nodrop, rustc-demangle, cfg-if, bumpalo,
stable_deref_trait
MIT (1): synstructure
MIT OR Apache-2.0 (9): once_cell, failure, log, quote, proc-macro2, syn, failure_derive, unicode-xid, libc
Unlicense OR MIT (1): byteorder

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
HarfBuzz is licensed under the so-called "Old MIT" license. Details follow. For parts of HarfBuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stibrley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Copyright (c) 2011 The Chromium OS Authors. All rights reserved.
Use of this source code is governed by a BSD-style license that can be found in the LICENSE file.
Copyright (c) 2015, Ilya Radchenko <ilya@burstcreations.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
   You must give any other recipients of the Work or Derivative Works a copy of this License; and
   You must cause any modified files to carry prominent notices stating that You changed the files; and
   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
   You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2011 Google Inc. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright (c) 2018 Google Inc. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright (c) 2015-2017 Rubn Norte <rubennorte@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

OGRE (www.ogre3d.org) is made available under the MIT License.

Copyright (c) 2000-2013 Torus Knot Software Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2011 The Chromium OS Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/**
 ** Copyright (c) 2007-2010 The Khronos Group Inc.
 **
 ** Permission is hereby granted, free of charge, to any person obtaining a
 ** copy of this software and/or associated documentation files (the
 ** "Materials"), to deal in the Materials without restriction, including
 ** without limitation the rights to use, copy, modify, merge, publish,
 ** distribute, sublicense, and/or sell copies of the Materials, and to
 ** permit persons to whom the Materials are furnished to do so, subject to
 ** the following conditions:
 
 Open Source Used In DNAC Platform Arica 5492
The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Qt port of geosimplify.js, https://github.com/mapbox/geosimplify-js

Copyright (c) 2017, Daniel Patterson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY 
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED 
WARRANTIES OF 
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT 
SHALL THE 
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, 
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT 
OF 
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS 
INTERRUPTION) 
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT 
LIABILITY, OR 
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS 
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---------------------------------
Based on simplify-js by Vladimir Agafonkin - http://mourner.github.io/simplify-js/
ISC License

Copyright (c) 2015, Mapbox

Permission to use, copy, modify, and/or distribute this software for any purpose 
with or without fee is hereby granted, provided that the above copyright notice 
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH 
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND 
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, 
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM 
LOSS 
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER 
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF 
THIS SOFTWARE.
The MIT License

Copyright (C) 2011-2013 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of 
this software and associated documentation files (the "Software"), to deal in 
the Software without restriction, including without limitation the rights to 
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies 
of the Software, and to permit persons to whom the Software is furnished to do 
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2017 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3. That license references the General Public License version 3, that is displayed below. Other portions of the Qt Toolkit may be licensed directly under this license.

-------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.
To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.
"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.
If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:
a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the
covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
Open Source Used In DNAC Platform Arica 5509

notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
Copyright (c) 2014-2019 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2012, 2013 Thorsten Lorenz <thlorenz@gmx.de>
Copyright (c) 2012 James Halliday <mail@substack.net>
Copyright (c) 2009 Thomas Robinson <280north.com>

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright 2016 Google Inc. All rights reserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016, Google Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
The MIT License (MIT)

Copyright (c) 2014 Cameron Lowell Palmer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===

Some of the benchmark data in testdata/ is licensed differently:

- fireworks.jpeg is Copyright 2013 Steinar H. Gunderson, and is licensed under the Creative Commons Attribution 3.0 license (CC-BY-3.0). See https://creativecommons.org/licenses/by/3.0/ for more information.

- kppkn.gtb is taken from the Gaviota chess tablebase set, and is licensed under the MIT License. See
https://sites.google.com/site/gaviotachessengine/Home/endgame-tablebases-1 for more information.

- paper-100k.pdf is an excerpt (bytes 92160 to 194560) from the paper Combinatorial Modeling of Chromatin Features Quantitatively Predicts DNA Replication Timing in _Drosophila_ by Federico Comoglio and Renato Paro, which is licensed under the CC-BY license. See http://www.ploscompbiol.org/static/license for more information.

- alice29.txt, asyoulik.txt, prlabn12.txt and lcet10.txt are from Project Gutenberg. The first three have expired copyrights and are in the public domain; the latter does not have expired copyright, but is still in the public domain according to the license information (http://www.gutenberg.org/ebooks/53).

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

qimagetransform.cpp was contributed by Daniel M. Duley based on code from Imlib2.

Copyright (C) 2004, 2005 Daniel M. Duley

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Imlib2 License

Copyright (C) 2000 Carsten Haitzler and various contributors (see
AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies of the Software and its Copyright notices. In addition
publicly documented acknowledgment must be given that this software has
been used if no source code of this software is made available publicly.
This includes acknowledgments in either Copyright notices, Manuals,
Publicity and Marketing documents or any documentation provided with any
product containing this software. This License does not apply to any
software that links to the libraries provided by this software
(statically or dynamically), but only to the software provided.

Please see the COPYING.PLAIN for a plain-english explanation of this
notice and it's intent.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

language: node_js
default:
- "6"
- "8"
Copyright (c) 2014-2018 Suguru Motegi
Based on Async.js, Copyright Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Extracted from mac/include/DeckLinkAPI.h:

** Copyright (c) 2014 Blackmagic Design
**
** Permission is hereby granted, free of charge, to any person or organization
** obtaining a copy of the software and accompanying documentation covered by
** this license (the "Software") to use, reproduce, display, distribute,
** execute, and transmit the Software, and to prepare derivative works of the
** Software, and to permit third-parties to whom the Software is furnished to ** do so, all subject to the following:
**
** The copyright notices in the Software and this entire statement, including ** the above license grant, this restriction and the following disclaimer, ** must be included in all copies of the Software, in whole or in part, and ** all derivative works of the Software, unless such copies or derivative ** works are solely in the form of machine-executable object code generated by ** a source language processor.
**
** THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR ** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, ** FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT ** SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE ** FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ** ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER ** DEALINGS IN THE SOFTWARE.
Copyright (c) 2012 Barnesandnoble.com, llc, Donavon West, and Domenic Denicola

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 1999-2000 Mizi Research Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 2.1, which is displayed below.

----------------------------------------------------------

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

**Preamble**

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.
Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the
b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUMED THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
A tiny JavaScript debugging utility modelled after Node.js core's debugging technique. Works in Node.js and web browsers.

## Installation

```bash
$ npm install debug
```

## Usage

`debug` exposes a function; simply pass this function the name of your module, and it will return a decorated version of `console.error` for you to pass debug statements to. This will allow you to toggle the debug output for different parts of your module as well as the module as a whole.

Example [_app.js_](./examples/node/app.js):

```js
 var debug = require('debug')('http')
 , http = require('http')
```
```javascript
.name = 'My App';

// fake app

debug('booting %o', name);

http.createServer(function(req, res){
    debug(req.method + ' ' + req.url);
    res.end('hello
');
}).listen(3000, function(){
    debug('listening');
});

// fake worker of some kind

require('./worker');
...

Example `_worker.js`(_/examples/node/worker.js):

```javascript
```
```
Windows command prompt notes

CMD

On Windows the environment variable is set using the `set` command.
```cmd
set DEBUG=*,not_this
```

Example:
```cmd
set DEBUG=*,not_this & node app.js
```

PowerShell (VS Code default)

PowerShell uses different syntax to set environment variables.
```cmd
$env:DEBUG = "*,not_this"
```

Example:
```cmd
$env:DEBUG='app';node app.js
```

Then, run the program to be debugged as usual.

npm script example:
```js
"windowsDebug": "@powershell -Command $env:DEBUG='*';node app.js",
```

Namespace Colors

Every debug instance has a color generated for it based on its namespace name. This helps when visually parsing the debug output to identify which debug instance a debug line belongs to.

Node.js
In Node.js, colors are enabled when stderr is a TTY. You also _should_ install the `supports-color` [module alongside debug](https://npmjs.org/supports-color), otherwise debug will only use a small handful of basic colors.

Colors are also enabled on "Web Inspectors" that understand the `%c` formatting option. These are WebKit web inspectors, Firefox ([since version 31](https://hacks.mozilla.org/2014/05/editable-box-model-multiple-selection-sublime-text-keys-much-more-firefox-developer-tools-episode-31/)) and the Firebug plugin for Firefox (any version).

When stdout is not a TTY, `Date#toISOString()` is used, making it more useful for logging the debug information as shown below:

If you're using this in one or more of your libraries, you _should_ use the name of your library so that developers may toggle debugging as desired without guessing names. If you have more than one debuggers you _should_ prefix them with your library name and use `:` to separate features. For example "bodyParser" from Connect would then be "connect:bodyParser". If you append a "+" to the end of your name, it will always be enabled regardless of the setting of the DEBUG environment variable. You can then use it for normal output as well as debug output.

The `*` character may be used as a wildcard. Suppose for example your library has debuggers named "connect:bodyParser", "connect:compress", "connect:session", instead of listing all three with
You can also exclude specific debuggers by prefixing them with a "-" character. For example, `DEBUG=-connect:*` would include all debuggers except those starting with "connect: ".

## Environment Variables

When running through Node.js, you can set a few environment variables that will change the behavior of the debug logging:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>DEBUG</code></td>
<td>Enables/disables specific debugging namespaces.</td>
</tr>
<tr>
<td><code>DEBUG_HIDE_DATE</code></td>
<td>Hide date from debug output (non-TTY).</td>
</tr>
<tr>
<td><code>DEBUG_COLORS</code></td>
<td>Whether or not to use colors in the debug output.</td>
</tr>
<tr>
<td><code>DEBUGDEPTH</code></td>
<td>Object inspection depth.</td>
</tr>
<tr>
<td><code>DEBUGSHOWHIDDEN</code></td>
<td>Shows hidden properties on inspected objects.</td>
</tr>
</tbody>
</table>

__Note__: The environment variables beginning with `DEBUG_` end up being converted into an Options object that gets used with `%o` / `%O` formatters. See the Node.js documentation for `util.inspect()` for the complete list.

## Formatters

Debug uses [printf-style](https://en.wikipedia.org/wiki/Printf_format_string) formatting. Below are the officially supported formatters:

<table>
<thead>
<tr>
<th>Formatter</th>
<th>Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>%O</code></td>
<td>Pretty-print an Object on multiple lines.</td>
</tr>
<tr>
<td><code>%o</code></td>
<td>Pretty-print an Object all on a single line.</td>
</tr>
<tr>
<td><code>%s</code></td>
<td>String.</td>
</tr>
<tr>
<td><code>%d</code></td>
<td>Number (both integer and float).</td>
</tr>
<tr>
<td><code>%j</code></td>
<td>JSON. Replaced with the string '[Circular]' if the argument contains circular references.</td>
</tr>
<tr>
<td><code>%%</code></td>
<td>Single percent sign ('%'). This does not consume an argument.</td>
</tr>
</tbody>
</table>

### Custom formatters

You can add custom formatters by extending the `debug.formatters` object. For example, if you wanted to add support for rendering a Buffer as hex with `%%h`, you could do something like:
```js
const createDebug = require('debug')
createDebug.formatters.h = (v) => {
  return v.toString('hex')
}

// elsewhere
const debug = createDebug('foo')
dep('this is hex: %h', new Buffer('hello world'))
// foo this is hex: 68656c6c6f20776f726c6421 +0ms
```
Example [_stdout.js_](./examples/node/stdout.js):

```js
var debug = require('debug');
var error = debug('app:error');

// by default stderr is used
error('goes to stderr!');

var log = debug('app:log');
// set this namespace to log via console.log
log.log = console.log.bind(console); // don't forget to bind to console!
log('goes to stdout');
error('still goes to stderr!');

// set all output to go via console.info
// overrides all per-namespace log settings
debug.log = console.info.bind(console);
error('now goes to stdout via console.info');
log('still goes to stdout, but via console.info now');
```

## Extend
You can simply extend debugger
```js
const log = require('debug')('auth');

//creates new debug instance with extended namespace
const logSign = log.extend('sign');
const logLogin = log.extend('login');

log('hello'); // auth hello
logSign('hello'); // auth:sign hello
logLogin('hello'); // auth:login hello
```

## Set dynamically
You can also enable debug dynamically by calling the `enable()` method:
```js
let debug = require('debug');

console.log(1, debug.enabled('test'));

debug.enable('test');

console.log(2, debug.enabled('test'));
```
debug.disable();
console.log(3, debug.enabled('test'));

```javascript

```
```

Usage :
`enable(namespaces)`
`namespaces` can include modes separated by a colon and wildcards.

Note that calling `enable()` completely overrides previously set DEBUG variable:

```bash
$ DEBUG=foo node -e 'var dbg = require("debug"); dbg.enable("bar"); console.log(dbg.enabled("foo"))'
```
```javascript

```bash
=> false
```
Support us with a monthly donation and help us continue our activities. [[Become a backer](https://opencollective.com/debug#backer)]

<a href="https://opencollective.com/debug/backer/0/website" target="_blank"><img src="https://opencollective.com/debug/backer/0/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/1/website" target="_blank"><img src="https://opencollective.com/debug/backer/1/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/2/website" target="_blank"><img src="https://opencollective.com/debug/backer/2/avatar.svg"></a>


<a href="https://opencollective.com/debug/backer/7/website" target="_blank"><img src="https://opencollective.com/debug/backer/7/avatar.svg"></a>


<a href="https://opencollective.com/debug/backer/10/website" target="_blank"><img src="https://opencollective.com/debug/backer/10/avatar.svg"></a>


<a href="https://opencollective.com/debug/backer/12/website" target="_blank"><img src="https://opencollective.com/debug/backer/12/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/13/website" target="_blank"><img src="https://opencollective.com/debug/backer/13/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/14/website" target="_blank"><img src="https://opencollective.com/debug/backer/14/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/15/website" target="_blank"><img src="https://opencollective.com/debug/backer/15/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/16/website" target="_blank"><img src="https://opencollective.com/debug/backer/16/avatar.svg"></a>


<a href="https://opencollective.com/debug/backer/18/website" target="_blank"><img src="https://opencollective.com/debug/backer/18/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/19/website" target="_blank"><img src="https://opencollective.com/debug/backer/19/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/20/website" target="_blank"><img src="https://opencollective.com/debug/backer/20/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/21/website" target="_blank"><img src="https://opencollective.com/debug/backer/21/avatar.svg"></a>

<a href="https://opencollective.com/debug/backer/22/website" target="_blank"><img src="https://opencollective.com/debug/backer/22/avatar.svg"></a>
## Sponsors

Become a sponsor and get your logo on our README on Github with a link to your site. [[Become a sponsor](https://opencollective.com/debug#sponsor)]
## License

(The MIT License)

Copyright (c) 2014-2017 TJ Holowaychuk &lt;tj@vision-media.ca&gt;:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License
Copyright (c) 2013-2018 Franck34

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ISC License
Copyright (c) 2016 Mapbox

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND ISC DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
MIT License

Copyright (C) 2010-2019 by Philipp Dunkel, Ben Noordhuis, Elan Shankar, Paul Miller

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Thiago de Arruda

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2017, The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
 REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
```
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

# The MIT License (MIT)

Copyright 2016-2019 Martin Mit

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (C) 2013 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
APPLE PUBLIC SOURCE LICENSE
Version 2.0 - August 6, 2003

Please read this License carefully before downloading this software. By downloading or using this software, you are agreeing to be bound by the terms of this License. If you do not or cannot agree to the terms of this License, please do not download or use the software.

Apple Note: In January 2007, Apple changed its corporate name from "Apple Computer, Inc." to "Apple Inc." This change has been reflected below and copyright years updated, but no other changes have been made to the APSL 2.0.

1. General; Definitions. This License applies to any program or other work which Apple Inc. ("Apple") makes publicly available and which contains a notice placed by Apple identifying such program or work as "Original Code" and stating that it is subject to the terms of this Apple Public Source License version 2.0 ("License"). As used in this License:

1.1 "Applicable Patent Rights" mean: (a) in the case where Apple is the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to Apple and (ii) that cover subject matter contained in the Original Code, but only to the extent necessary to use, reproduce and/or distribute the Original Code without infringement; and (b) in the case where You are the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to You and (ii) that cover subject matter in Your Modifications, taken alone or in combination with Original Code.

1.2 "Contributor" means any person or entity that creates or contributes to the creation of Modifications.

1.3 "Covered Code" means the Original Code, Modifications, the combination of Original Code and any Modifications, and/or any respective portions thereof.

1.4 "Externally Deploy" means: (a) to sublicense, distribute or otherwise make Covered Code available, directly or indirectly, to anyone other than You; and/or (b) to use Covered Code, alone or as part of a Larger Work, in any way to provide a service, including but not limited to delivery of content, through electronic communication with a client other than You.

1.5 "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.6 "Modifications" mean any addition to, deletion from, and/or change to,
the substance and/or structure of the Original Code, any previous Modifications, the combination of Original Code and any previous Modifications, and/or any respective portions thereof. When code is released as a series of files, a Modification is: (a) any addition to or deletion from the contents of a file containing Covered Code; and/or (b) any new file or other representation of computer program statements that contains any part of Covered Code.

1.7 "Original Code" means (a) the Source Code of a program or other work as originally made available by Apple under this License, including the Source Code of any updates or upgrades to such programs or works made available by Apple under this License, and that has been expressly identified by Apple as such in the header file(s) of such work; and (b) the object code compiled from such Source Code and originally made available by Apple under this License.

1.8 "Source Code" means the human readable form of a program or other work that is suitable for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an executable (object code).

1.9 "You" or "Your" means an individual or a legal entity exercising rights under this License. For legal entities, "You" or "Your" includes any entity which controls, is controlled by, or is under common control with, You, where "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of fifty percent (50%) or more of the outstanding shares or beneficial ownership of such entity.

2. Permitted Uses; Conditions & Restrictions. Subject to the terms and conditions of this License, Apple hereby grants You, effective on the date You accept this License and download the Original Code, a world-wide, royalty-free, non-exclusive license, to the extent of Apple's Applicable Patent Rights and copyrights covering the Original Code, to do the following:

2.1 Unmodified Code. You may use, reproduce, display, perform, internally distribute within Your organization, and Externally Deploy verbatim, unmodified copies of the Original Code, for commercial or non-commercial purposes, provided that in each instance:

(a) You must retain and reproduce in all copies of Original Code the copyright and other proprietary notices and disclaimers of Apple as they appear in the Original Code, and keep intact all notices in the Original Code that refer to this License; and

(b) You must include a copy of this License with every copy of Source Code of Covered Code and documentation You distribute or Externally Deploy, and You may not offer or impose any terms on such Source Code that alter or restrict this License or the recipients' rights hereunder, except as permitted under Section 6.
2.2 Modified Code. You may modify Covered Code and use, reproduce, display, perform, internally distribute within Your organization, and Externally Deploy Your Modifications and Covered Code, for commercial or non-commercial purposes, provided that in each instance You also meet all of these conditions:

(a) You must satisfy all the conditions of Section 2.1 with respect to the Source Code of the Covered Code;

(b) You must duplicate, to the extent it does not already exist, the notice in Exhibit A in each file of the Source Code of all Your Modifications, and cause the modified files to carry prominent notices stating that You changed the files and the date of any change; and

(c) If You Externally Deploy Your Modifications, You must make Source Code of all Your Externally Deployed Modifications either available to those to whom You have Externally Deployed Your Modifications, or publicly available. Source Code of Your Externally Deployed Modifications must be released under the terms set forth in this License, including the license grants set forth in Section 3 below, for as long as you Externally Deploy the Covered Code or twelve (12) months from the date of initial External Deployment, whichever is longer. You should preferably distribute the Source Code of Your Externally Deployed Modifications electronically (e.g. download from a web site).

2.3 Distribution of Executable Versions. In addition, if You Externally Deploy Covered Code (Original Code and/or Modifications) in object code, executable form only, You must include a prominent notice, in the code itself as well as in related documentation, stating that Source Code of the Covered Code is available under the terms of this License with information on how and where to obtain such Source Code.

2.4 Third Party Rights. You expressly acknowledge and agree that although Apple and each Contributor grants the licenses to their respective portions of the Covered Code set forth herein, no assurances are provided by Apple or any Contributor that the Covered Code does not infringe the patent or other intellectual property rights of any other entity. Apple and each Contributor disclaim any liability to You for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, You hereby assume sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow You to distribute the Covered Code, it is Your responsibility to acquire that license before distributing the Covered Code.

3. Your Grants. In consideration of, and as a condition to, the licenses granted to You under this License, You hereby grant to any person or entity receiving or distributing Covered Code under this License a non-exclusive,
royalty-free, perpetual, irrevocable license, under Your Applicable Patent Rights and other intellectual property rights (other than patent) owned or controlled by You, to use, reproduce, display, perform, modify, sublicense, distribute and Externally Deploy Your Modifications of the same scope and extent as Apple's licenses under Sections 2.1 and 2.2 above.

4. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In each such instance, You must make sure the requirements of this License are fulfilled for the Covered Code or any portion thereof.

5. Limitations on Patent License. Except as expressly stated in Section 2, no other patent rights, express or implied, are granted by Apple herein. Modifications and/or Larger Works may require additional patent licenses from Apple which Apple may grant in its sole discretion.

6. Additional Terms. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations and/or other rights consistent with the scope of the license granted herein ("Additional Terms") to one or more recipients of Covered Code. However, You may do so only on Your own behalf and as Your sole responsibility, and not on behalf of Apple or any Contributor. You must obtain the recipient's agreement that any such Additional Terms are offered by You alone, and You hereby agree to indemnify, defend and hold Apple and every Contributor harmless for any liability incurred by or claims asserted against Apple or such Contributor by reason of any such Additional Terms.

7. Versions of the License. Apple may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Once Original Code has been published under a particular version of this License, You may continue to use it under the terms of that version. You may also choose to use such Original Code under the terms of any subsequent version of this License published by Apple. No one other than Apple has the right to modify the terms applicable to Covered Code created under this License.

8. NO WARRANTY OR SUPPORT. The Covered Code may contain in whole or in part pre-release, untested, or not fully tested works. The Covered Code may contain errors that could cause failures or loss of data, and may be incomplete or contain inaccuracies. You expressly acknowledge and agree that use of the Covered Code, or any portion thereof, is at Your sole and entire risk. THE COVERED CODE IS PROVIDED "AS IS" AND WITHOUT WARRANTY, UPGRADES OR SUPPORT OF ANY KIND AND APPLE AND APPLE'S LICENSOR(S) (COLLECTIVELY REFERRED TO AS "APPLE" FOR THE PURPOSES OF SECTIONS 8 AND 9) AND ALL CONTRIBUTORS EXPRESSLY DISCLAIM ALL WARRANTIES AND/OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY, OF SATISFACTORY QUALITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY, OF
QUIET ENJOYMENT, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. APPLE AND EACH CONTRIBUTOR DOES NOT WARRANT AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE COVERED CODE, THAT THE FUNCTIONS CONTAINED IN THE COVERED CODE WILL MEET YOUR REQUIREMENTS, THAT THE OPERATION OF THE COVERED CODE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE COVERED CODE WILL BE CORRECTED. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY APPLE, AN APPLE AUTHORIZED REPRESENTATIVE OR ANY CONTRIBUTOR SHALL CREATE A WARRANTY. You acknowledge that the Covered Code is not intended for use in the operation of nuclear facilities, aircraft navigation, communication systems, or air traffic control machines in which case the failure of the Covered Code could lead to death, personal injury, or severe physical or environmental damage.

9. LIMITATION OF LIABILITY. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT SHALL APPLE OR ANY CONTRIBUTOR BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE OR YOUR USE OR INABILITY TO USE THE COVERED CODE, OR ANY PORTION THEREOF, WHETHER UNDER A THEORY OF CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCTS LIABILITY OR OTHERWISE, EVEN IF APPLE OR SUCH CONTRIBUTOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF ESSENTIAL PURPOSE OF ANY REMEDY. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF LIABILITY OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS LIMITATION MAY NOT APPLY TO YOU. In no event shall Apple's total liability to You for all damages (other than as may be required by applicable law) under this License exceed the amount of fifty dollars ($50.00).

10. Trademarks. This License does not grant any rights to use the trademarks or trade names "Apple", "Mac", "Mac OS", "QuickTime", "QuickTime Streaming Server" or any other trademarks, service marks, logos or trade names belonging to Apple (collectively "Apple Marks") or to any trademark, service mark, logo or trade name belonging to any Contributor. You agree not to use any Apple Marks in or as part of the name of products derived from the Original Code or to endorse or promote products derived from the Original Code other than as expressly permitted by and in strict compliance at all times with Apple's third party trademark usage guidelines which are posted at http://www.apple.com/legal/guidelinesfor3rdparties.html.

11. Ownership. Subject to the licenses granted under this License, each Contributor retains all rights, title and interest in and to any Modifications made by such Contributor. Apple retains all rights, title and interest in and to the Original Code and any Modifications made by or on behalf of Apple ("Apple Modifications"), and such Apple Modifications will not be automatically subject to this License. Apple may, at its sole discretion, choose to license such Apple Modifications under this License, or on different terms from those contained in this License or may choose not to license them at all.

12. Termination.

12.1 Termination. This License and the rights granted hereunder will
terminate:

(a) automatically without notice from Apple if You fail to comply with any term(s) of this License and fail to cure such breach within 30 days of becoming aware of such breach;
(b) immediately in the event of the circumstances described in Section 13.5(b); or
(c) automatically without notice from Apple if You, at any time during the term of this License, commence an action for patent infringement against Apple; provided that Apple did not first commence an action for patent infringement against You in that instance.

12.2 Effect of Termination. Upon termination, You agree to immediately stop any further use, reproduction, modification, sublicensing and distribution of the Covered Code. All sublicenses to the Covered Code which have been properly granted prior to termination shall survive any termination of this License. Provisions which, by their nature, should remain in effect beyond the termination of this License shall survive, including but not limited to Sections 3, 5, 8, 9, 10, 11, 12.2 and 13. No party will be liable to any other for compensation, indemnity or damages of any sort solely as a result of terminating this License in accordance with its terms, and termination of this License will be without prejudice to any other right or remedy of any party.


13.1 Government End Users. The Covered Code is a "commercial item" as defined in FAR 2.101. Government software and technical data rights in the Covered Code include only those rights customarily provided to the public as defined in this License. This customary commercial license in technical data and software is provided in accordance with FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for Department of Defense purchases, DFAR 252.227-7015 (Technical Data -- Commercial Items) and 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation). Accordingly, all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

13.2 Relationship of Parties. This License will not be construed as creating an agency, partnership, joint venture or any other form of legal association between or among You, Apple or any Contributor, and You will not represent to the contrary, whether expressly, by implication, appearance or otherwise.

13.3 Independent Development. Nothing in this License will impair Apple's right to acquire, license, develop, have others develop for it, market and/or distribute technology or products that perform the same or similar functions as, or otherwise compete with, Modifications, Larger Works, technology or products that You may develop, produce, market or distribute.
13.4 Waiver; Construction. Failure by Apple or any Contributor to enforce any provision of this License will not be deemed a waiver of future enforcement of that or any other provision. Any law or regulation which provides that the language of a contract shall be construed against the drafter will not apply to this License.

13.5 Severability. (a) If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of the License will be enforced to the maximum extent permissible so as to effect the economic benefits and intent of the parties, and the remainder of this License will continue in full force and effect. (b) Notwithstanding the foregoing, if applicable law prohibits or restricts You from fully and/or specifically complying with Sections 2 and/or 3 or prevents the enforceability of either of those Sections, this License will immediately terminate and You must immediately discontinue any use of the Covered Code and destroy all copies of it that are in your possession or control.

13.6 Dispute Resolution. Any litigation or other dispute resolution between You and Apple relating to this License shall take place in the Northern District of California, and You and Apple hereby consent to the personal jurisdiction of, and venue in, the state and federal courts within that District with respect to this License. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded.

13.7 Entire Agreement; Governing Law. This License constitutes the entire agreement between the parties with respect to the subject matter hereof. This License shall be governed by the laws of the United States and the State of California, except that body of California law concerning conflicts of law.

Where You are located in the province of Quebec, Canada, the following clause applies: The parties hereby confirm that they have requested that this License and all related documents be drafted in English. Les parties ont exig que le present contrat et tous les documents connexes soient rdigs en anglais.

EXHIBIT A.

"Portions Copyright (c) 1999-2007 Apple Inc. All Rights Reserved.

This file contains Original Code and/or Modifications of Original Code as defined in and that are subject to the Apple Public Source License Version 2.0 (the 'License'). You may not use this file except in compliance with the License. Please obtain a copy of the License at http://www.opensource.apple.com/apsl/ and read it before using this file.

The Original Code and all software distributed under the License are distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT
Mozilla Public License Version 2.0

1. Definitions

1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
version 1.1 or earlier of the License, but not also under the
terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation
Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered
Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.
*                                                                      *
************************************************************************
************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort
8. Litigation

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions
You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
The MIT License (MIT)

Copyright (c) 2014 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

****************************************************************************
**
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
**
** This file is part of the FOO module of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:LGPL$
** Commercial License Usage
** Licensees holding valid commercial Qt licenses may use this file in accordance with the commercial license agreement provided with the Software or, alternatively, in accordance with the terms contained in a written agreement between you and The Qt Company. For licensing terms and conditions see https://www.qt.io/terms-conditions. For further information use the contact form at https://www.qt.io/contact-us.
**
** GNU Lesser General Public License Usage
** Alternatively, this file may be used under the terms of the GNU Lesser General Public License version 3 as published by the Free Software Foundation and appearing in the file LICENSE.LGPL3 included in the packaging of this file. Please review the following information to ensure the GNU Lesser General Public License version 3 requirements will be met: https://www.gnu.org/licenses/lgpl-3.0.html.
**
** GNU General Public License Usage
** Alternatively, this file may be used under the terms of the GNU General Public License version 2.0 or (at your option) the GNU General Public license version 3 or any later version approved by the KDE Free Qt Foundation. The licenses are as published by the Free Software Foundation and appearing in the file LICENSE.GPL2 and LICENSE.GPL3 included in the packaging of this file. Please review the following information to ensure the GNU General Public License requirements will be met: https://www.gnu.org/licenses/gpl-2.0.html and https://www.gnu.org/licenses/gpl-3.0.html.
**
All MurmurHash source files are placed in the public domain.

The license below applies to all other code in SMHasher:

Copyright (c) 2011 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2017, Mozilla
Copyright (c) 2007-2017, Jean-Marc Valin
Copyright (c) 2005-2017, Xiph.Org Foundation
Copyright (c) 2003-2004, Mark Borgerding

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.Org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
(The MIT License)

Copyright (c) 2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2011-2017 JP Richardson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files
(the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that
You changed the files; and
You must retain, in the Source form of any Derivative Works that You
distribute, all copyright, patent, trademark, and attribution notices
from the Source form of the Work, excluding those notices that do not
pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution,
then any Derivative Works that You distribute must include a readable
copy of the attribution notices contained within such NOTICE file, excluding
those notices that do not pertain to any part of the Derivative Works, in
at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if
provided along with the Derivative Works; or, within a display generated by
the Derivative Works, if and wherever such third-party notices normally
appear. The contents of the NOTICE file are for informational purposes
only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as
an addendum to the NOTICE text from the Work, provided that such additional
attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide
additional or different license terms and conditions for use, reproduction, or
distribution of Your modifications, or for any such Derivative Works as a
whole, provided Your use, reproduction, and distribution of the Work otherwise
complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any
Contribution intentionally submitted for inclusion in the Work by You to the
Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein
shall supersede or modify the terms of any separate license agreement you
may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names,
trademarks, service marks, or product names of the Licensor, except as
required for reasonable and customary use in describing the origin of the
Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to
in writing, Licensor provides the Work (and each Contributor provides its
Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
ANY KIND, either express or implied, including, without limitation, any
warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or
FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining
the appropriateness of using or redistributing the Work and assume any risks
associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in
tort (including negligence), contract, or otherwise, unless required by
applicable law (such as deliberate and grossly negligent acts) or agreed to
in writing, shall any Contributor be liable to You for damages, including
any direct, indirect, special, incidental, or consequential damages of any
character arising as a result of this License or out of the use or inability
to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other
commercial damages or losses), even if such Contributor has been advised
of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the
Work or Derivative Works thereof, You may choose to offer, and charge a
fee for, acceptance of support, warranty, indemnity, or other liability
obligations and/or rights consistent with this License. However, in accepting
such obligations, You may act only on Your own behalf and on Your sole
responsibility, not on behalf of any other Contributor, and only if You
agree to indemnify, defend, and hold each Contributor harmless for any
liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
# MIT LICENSED Copyright (c) 2013 Arnout Kazemier (http://3rd-Eden.com)
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
# THE SOFTWARE.
Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software
for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice
appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

GNU Free Documentation License
Version 1.3, 3 November 2008

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License
principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available
The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the
copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the
4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Open Source Used In DNAC Platform Arica 5590
Preserve the Title of the section, and preserve in the section all
the substance and tone of each of the contributor acknowledgements
and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document,
unaltered in their text and in their titles. Section numbers
or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section
may not be included in the Modified Version.

N. Do not retile any existing section to be Entitled "Endorsements"
or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or
appendices that qualify as Secondary Sections and contain no material
copied from the Document, you may at your option designate some or all
of these sections as invariant. To do this, add their titles to the
list of Invariant Sections in the Modified Version's license notice.
These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains
nothing but endorsements of your Modified Version by various
parties--for example, statements of peer review or that the text has
been approved by an organization as the authoritative definition of a
standard.

You may add a passage of up to five words as a Front-Cover Text, and a
passage of up to 25 words as a Back-Cover Text, to the end of the list
of Cover Texts in the Modified Version. Only one passage of
Front-Cover Text and one of Back-Cover Text may be added by (or
through arrangements made by) any one entity. If the Document already
includes a cover text for the same cover, previously added by you or
by arrangement made by the same entity you are acting on behalf of,
you may not add another; but you may replace the old one, on explicit
permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License
give permission to use their names for publicity for or to assert or
imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this
License, under the terms defined in section 4 above for modified
versions, provided that you include in the combination all of the
Invariant Sections of all of the original documents, unmodified, and
list them all as Invariant Sections of your combined work in its
license notice, and that you preserve all their Warranty Disclaimers.
The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the
8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or
in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

MIT License
Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Julien Pommier (pommier@modartt.com)

Based on original fortran 77 code from FFTPACKv4 from NETLIB, authored by Dr Paul Swarztrauber of NCAR, in 1985.

As confirmed by the NCAR fftpack software curators, the following FFTPACKv5 license applies to FFTPACKv4 sources. My changes are released under the same terms.

FFTPACK license:

http://www.cisl.ucar.edu/css/software/fftpack5/ftpk.html

Copyright (c) 2004 the University Corporation for Atmospheric Research ("UCAR"). All rights reserved. Developed by NCAR's Computational and Information Systems Laboratory, UCAR, www.cisl.ucar.edu.

Redistribution and use of the Software in source and binary forms, with or without modification, is permitted provided that the following conditions are met:

- Neither the names of NCAR's Computational and Information Systems Laboratory, the University Corporation for Atmospheric Research, nor the names of its sponsors or contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

- Redistributions of source code must retain the above copyright notices, this list of conditions, and the disclaimer below.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer below in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

-----------------------------------------------------------

PREAMBLE

The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS

Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to
remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

The FreeType 2 font engine is copyrighted work and cannot be used legally without a software license. In order to make this project usable to a vast majority of developers, we distribute it under two mutually exclusive open-source licenses.

This means that *you* must choose *one* of the two licenses described below, then obey all its terms and conditions when using FreeType 2 in any of your projects or products.

- The FreeType License, found in the file `FTL.TXT', which is similar to the original BSD license *with* an advertising clause that forces you to explicitly cite the FreeType project in your product's documentation. All details are in the license file. This license is suited to products which don't use the GNU General Public License.

  Note that this license is compatible to the GNU General Public License version 3, but not version 2.

- The GNU General Public License version 2, found in `GPLv2.TXT' (any later version can be used also), for programs which already use the GPL. Note that the FTL is incompatible with GPLv2 due to its advertisement clause.

The contributed BDF and PCF drivers come with a license similar to that of the X Window System. It is compatible to the above two licenses (see file src/bdf/README and src/pcf/README). The same holds for the files `fthash.c' and `fthash.h'; their code was part of the BDF driver in earlier FreeType versions.
The gzip module uses the zlib license (see src/gzip/zlib.h) which too is compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds) is in the public domain.

--- end of LICENSE.TXT ---

/*
 * SpanDSP - a series of DSP components for telephony
 *
 * g711.h - In line A-law and u-law conversion routines
 *
 * Written by Steve Underwood <steveu@coppice.org>
 *
 * Copyright (C) 2001 Steve Underwood
 *
 * Despite my general liking of the GPL, I place this code in the public domain for the benefit of all mankind - even the slimy ones who might try to proprietize my work and use it to my detriment.
 */

Copyright (c) 2006, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
Copyright 2001-2004 Unicode, Inc.

Disclaimer

This source code is provided as is by Unicode, Inc. No claims are made as to fitness for any particular purpose. No warranties of any kind are expressed or implied. The recipient agrees to determine applicability of information provided. If this file has been purchased on magnetic or optical media from Unicode, Inc., the sole remedy for any claim will be exchange of defective media within 90 days of receipt.

Limitations on Rights to Redistribute This Code

Unicode, Inc. hereby grants the right to freely use the information supplied in this file in the creation of products supporting the Unicode Standard, and to make copies of this file in any form for internal or external distribution as long as this notice remains attached.

# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# Names should be added to this file as:
#   Name <email address>

Mark Mentovai <mark@chromium.org>
Robert Sesek <rsesek@chromium.org>
Scott Graham <scottmg@chromium.org>
Joshua Peraza <jperaza@chromium.org>
SMSLib Sudden Motion Sensor Access Library

Copyright (c) 2010 Suitable Systems
All rights reserved.

Developed by: Daniel Griscom
Suitable Systems
http://www.suitable.com

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"), to deal with the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.

- Neither the names of Suitable Systems nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

For more information about SMSLib, see 
<http://www.suitable.com/tools/smslib.html>
or contact
Daniel Griscom
Suitable Systems
1 Centre Street, Suite 204
Wakefield, MA 01880
(781) 665-0053
COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2019 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
Open Source Used In DNAC Platform Arica 5604

OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY
SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

All trademarks and registered trademarks mentioned herein are the
property of their respective owners.


    # The Google Chrome software developed by Google is licensed under
    # the BSD license. Other software included in this distribution is
    # provided under other licenses, as set forth below.
    #
    # The BSD License
    # http://opensource.org/licenses/bsd-license.php
    # Copyright (C) 2006-2008, Google Inc.
    #
    # All rights reserved.
    #
    # Redistribution and use in source and binary forms, with or without
    # modification, are permitted provided that the following conditions are met:
    #
    # Redistributions of source code must retain the above copyright notice,
    # this list of conditions and the following disclaimer.
    # Redistributions in binary form must reproduce the above
    # copyright notice, this list of conditions and the following
    # disclaimer in the documentation and/or other materials provided with
    # the distribution.
    # Neither the name of Google Inc. nor the names of its
    # contributors may be used to endorse or promote products derived from
    # this software without specific prior written permission.
    #
    #
    # THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
    # CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
    # INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
    # MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
    # DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
    # LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
    # CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF

Open Source Used In DNAC Platform Arica 5604
# SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
# BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
# LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
# SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
# The word list in cjdict.txt are generated by combining three word lists
# listed below with further processing for compound word breaking. The
# frequency is generated with an iterative training against Google web
# corpora.
#
# * Libtabe (Chinese)
# - https://sourceforge.net/project/?group_id=1519
# - Its license terms and conditions are shown below.
#
# * IPADIC (Japanese)
# - http://chasen.aist-nara.ac.jp/chasen/distribution.html
# - Its license terms and conditions are shown below.
#
# --------COPYING.libtabe ---- BEGIN----------------------
#
# /*
# * Copyright (c) 1999 TaBE Project.
# * Copyright (c) 1999 Pai-Hsiang Hsiao.
# * All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * . Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * . Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * . Neither the name of the TaBE Project nor the names of its
# * contributors may be used to endorse or promote products derived
# * from this software without specific prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

Open Source Used In DNAC Platform Arica  5605
/* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)*/

# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.

*/
#
#
/*
 * Copyright (c) 1999 Computer Systems and Communication Lab,
 * Institute of Information Science, Academia
 * Sinica. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * . Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in
 *   the documentation and/or other materials provided with the
 *   distribution.
 * . Neither the name of the Computer Systems and Communication Lab
 *   nor the names of its contributors may be used to endorse or
 *   promote products derived from this software without specific
 *   prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu http://casper.beckman.uiuc.edu/~c-tsai4
#
# COPYING.libtabe----END--------------------------------

Use, reproduction, and distribution of this software is permitted. Any copy of this software, whether in its original form or modified, must include both the above copyright notice and the following paragraphs.

Nara Institute of Science and Technology (NAIST), the copyright holders, disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness, in no event shall NAIST be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortuous action, arising out of or in connection with the use or performance of this software.

A large portion of the dictionary entries originate from ICOT Free Software. The following conditions for ICOT Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its original form or modified, to any third party or parties, PROVIDED that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear on, or be attached to, the Program, which is distributed substantially in the same form as set out herein and that such intended distribution, if actually made, will neither violate or otherwise contravene any of the laws and regulations of the countries having jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to have agreed and understood, that there is no warranty whatsoever for the program and, accordingly, the entire risk arising from or otherwise connected with the program is assumed by the user.
Therefore, neither ICOT, the copyright holder, or any other organization that participated in or was otherwise related to the development of the program and their respective officials, directors, officers and other employees shall be held liable for any and all damages, including, without limitation, general, special, incidental and consequential damages, arising out of or otherwise in connection with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

------------COPYING.ipadic-----END----------------------------------

3. Lao Word Break Dictionary Data (laodict.txt)

Copyright (c) 2013 International Business Machines Corporation and others. All Rights Reserved.

# Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
#  (copied below)
#
# This file is derived from the above dictionary, with slight modifications.
# -----------------------------------------------------------------------------
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without modification,
# are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in
# binary form must reproduce the above copyright notice, this list of
# conditions and the following disclaimer in the documentation and/or
# other materials provided with the distribution.
#
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
#
# 4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
# github.com/kanyawtech/myanmar-karen-word-lists
#
# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
# with the distribution.
#
# Neither the name Myanmar Karen Word Lists, nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

# 7. Database Ownership

# The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

6. Google double-conversion

Copyright 2006-2011, the V8 project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided
with the distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The FreeType 2 font engine is copyrighted work and cannot be used
legally without a software license. In order to make this project
usable to a vast majority of developers, we distribute it under two
mutually exclusive open-source licenses.

This means that *you* must choose *one* of the two licenses described
below, then obey all its terms and conditions when using FreeType 2 in
any of your projects or products.

- The FreeType License, found in the file `FTL.TXT', which is similar
to the original BSD license *with* an advertising clause that forces
you to explicitly cite the FreeType project in your product's
documentation. All details are in the license file. This license
is suited to products which don't use the GNU General Public
License.

  Note that this license is compatible to the GNU General Public
  License version 3, but not version 2.

- The GNU General Public License version 2, found in `GPLv2.TXT' (any
  later version can be used also), for programs which already use the
  GPL. Note that the FTL is incompatible with GPLv2 due to its
  advertisement clause.

The contributed BDF and PCF drivers come with a license similar to that
of the X Window System. It is compatible to the above two licenses (see
file src/bdf/README and src/pcf/README).

The gzip module uses the zlib license (see src/gzip/zlib.h) which too is
compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds)
is in the public domain.

--- FDL.TXT ---

The FreeType Project LICENSE
-----------------------------

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction
============

The FreeType Project is distributed in several archive packages;
some of them may contain, in addition to the FreeType font engine,
various tools and contributions which rely on, or relate to, the
FreeType Project.

This license applies to all files found in such packages, and
which do not fall under their own explicit license. The license
affects thus the FreeType font engine, the test programs,
documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG
(Independent JPEG Group) licenses, which all encourage inclusion
and use of free software in commercial and freeware products
alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be
  interested in any kind of bug reports. ('as is' distribution)

- You can use this software for whatever you want, in parts or
  full form, without having to pay us. ('royalty-free' usage)

- You may not pretend that you wrote this software. If you use
  it, or only parts of it, in a program, you must acknowledge
  somewhere in your documentation that you have used the
  FreeType code. ('credits')

We specifically permit and encourage the inclusion of this
software, with or without modifications, in commercial products.
We disclaim all warranties covering The FreeType Project and
assume no liability related to The FreeType Project.
Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```markdown

Portions of this software are copyright <year> The FreeType Project (www.freetype.org). All rights reserved.
```

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms
============

0. Definitions
------------

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty
------------

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to
accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts
-----------

There are two mailing lists related to FreeType:

- freetype@nongnu.org
  Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

- freetype-devel@nongnu.org
  Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

http://www.freetype.org

--- end of FDL.TXT ---

--- GPLv2.TXT ---

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

--- end of GPLv2.TXT ---
Font license info

## Font Awesome

Copyright (C) 2016 by Dave Gandy

Author: Dave Gandy
License: SIL ()
Homepage: http://fortawesome.github.com/Font-Awesome/
Copyright (c) 2015, Google Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

====================
Open Source Licenses
====================
This software may use portions of the following libraries subject to the accompanying licenses:

**********************
chromium_audio
**********************
// Copyright 2014 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**********************
curl
**********************
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2014, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright
notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

***************************************************************
dynamic_annotations
***************************************************************
Copyright (c) 2008-2009, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

***************************************************************
eigen3
Eigen is primarily MPL2 licensed. See COPYING.MPL2 and these links:
http://www.mozilla.org/MPL/2.0/
http://www.mozilla.org/MPL/2.0/FAQ.html

Some files contain third-party code under BSD or LGPL licenses, whence
the other COPYING.* files here.

All the LGPL code is either LGPL 2.1-only, or LGPL 2.1-or-later.
For this reason, the COPYING.LGPL file contains the LGPL 2.1 text.

If you want to guarantee that the Eigen code that you are #including
is licensed under the MPL2 and possibly more permissive licenses (like
BSD), #define this preprocessor symbol: EIGEN_MPL2_ONLY
For example, with most compilers, you could add this to your project
CXXFLAGS: -DEIGEN_MPL2_ONLY
This will cause a compilation error to be generated if you #include
any code that is LGPL licensed.

Following applies to:
./test/mapstaticmethods.cpp
./test/schur_real.cpp
./test/prec_inverse_4x4.cpp
./test/smallvectors.cpp
./test/redux.cpp
./test/special_numbers.cpp
./test/adjoint.cpp
./test/resize.cpp
./test/mixingtypes.cpp
./test/product_trmv.cpp
./test/sparse_solvers.cpp
./test/cholesky.cpp
./test/geo_quaternion.cpp
./test/miscmatrices.cpp
./test/stddeque.cpp
./test/integer_types.cpp
./test/product_large.cpp
./test/eigensolver_generic.cpp
./test/householder.cpp
./test/geo_orthomethods.cpp
./test/array_for_matrix.cpp
./test/sparseLM.cpp
./test/upperbidiagonalization.cpp
./test/nomalloc.cpp
./test/packetmath.cpp
./test/jacobisvd.cpp
./test/geo_transformations.cpp
/test/selfadjoint.cpp
/test/permutationmatrices.cpp
/test/superlu_support.cpp
/test/qtvector.cpp
/test/geo_homogeneous.cpp
/test/determinant.cpp
/test/array_reverse.cpp
/test/unalignedassert.cpp
/test/stdlist.cpp
/test/product_symm.cpp
/test/corners.cpp
/test/dontalign.cpp
/test/visitor.cpp
/test/geo_alignedbox.cpp
/test/diagonalmatrices.cpp
/test/product_small.cpp
/test/eigensolver_generalized_real.cpp
/test/umfpack_support.cpp
/test/first_aligned.cpp
/test/qr_fullpivoting.cpp
/test/array_replicate.cpp
/test/geoparametrizedline.cpp
/test/eigen2/eigen2_unalignedassert.cpp
/test/eigen2/eigen2_prec_inverse_4x4.cpp
/test/eigen2/eigen2_alignedbox.cpp
/test/eigen2/eigen2_sparse_product.cpp
/test/eigen2/eigen2_meta.cpp
/test/eigen2/eigen2_nomalloc.cpp
/test/eigen2/eigen2_visitor.cpp
/test/eigen2/eigen2_packetmath.cpp
/test/eigen2/eigen2_svd.cpp
/test/eigen2/eigen2_mixingtypes.cpp
/test/eigen2/eigen2_qr.cpp
/test/eigen2/eigen2_cwiscop.cpp
/test/eigen2/eigen2_geometry_with_eigen2_prefix.cpp
/test/eigen2/eigen2_smallvectors.cpp
/test/eigen2/eigen2_commainitializer.cpp
/test/eigen2/eigen2_sparse_solvers.cpp
/test/eigen2/eigen2_hyperplane.cpp
/test/eigen2/eigen2_eigensolver.cpp
/test/eigen2/eigen2_linearstructure.cpp
/test/eigen2/eigen2_sizeof.cpp
/test/eigen2/eigen2_parametrizedline.cpp
/test/eigen2/eigen2_lu.cpp
/test/eigen2/eigen2_adjoint.cpp
/test/eigen2/eigen2_geometry.cpp
/test/eigen2/eigen2_stdvector.cpp
/test/eigen2/eigen2_newstdvector.cpp
Open Source Used In DNAC Platform Arica 5634

ifecycle

/unsupported/Eigen/src/IterativeSolvers/IncompleteCholesky.h
/unsupported/Eigen/src/IterativeSolvers/Scaling.h
/unsupported/Eigen/src/IterativeSolvers/MINRES.h
/unsupported/Eigen/src/SparseExtra/RandomSetter.h
/unsupported/Eigen/src/SparseExtra/MatrixMarketIterator.h
/unsupported/Eigen/src/SparseExtra/DynamicSparseMatrix.h
/unsupported/Eigen/src/SparseExtra/MarketIO.h
/unsupported/Eigen/src/SparseExtra/BlockOfDynamicSparseMatrix.h
/unsupported/Eigen/src/KroneckerProduct/KroneckerTensorProduct.h
/unsupported/Eigen/src/NonLinearOptimization/LevenbergMarquardt.h
/unsupported/Eigen/src/NonLinearOptimization/HybridNonLinearSolver.h
/unsupported/Eigen/src/BVH/BVAlgorithms.h
/unsupported/Eigen/src/BVH/KdBVH.h
/unsupported/Eigen/src/AutoDiff/AutoDiffScalar.h
/unsupported/Eigen/src/AutoDiff/AutoDiffJacobian.h
/unsupported/Eigen/src/AutoDiff/AutoDiffVector.h
/unsupported/Eigen/src/Splines/Spline.h
/unsupported/Eigen/src/Splines/SplineFitting.h
/unsupported/Eigen/src/Splines/SplineFwd.h
/unsupported/Eigen/src/SVD/JacobiSVD.h
/unsupported/Eigen/src/SVD/BCSVD.h
/unsupported/Eigen/src/SVD/SVDBase.h
/unsupported/Eigen/src/MatrixFunctions/MatrixFunction.h
/unsupported/Eigen/src/MatrixFunctions/MatrixSquareRoot.h
/unsupported/Eigen/src/MatrixFunctions/MatrixLogarithm.h
/unsupported/Eigen/src/MatrixFunctions/StemFunction.h
/unsupported/Eigen/src/MatrixFunctions/MatrixPower.h
/unsupported/Eigen/src/MatrixFunctions/MatrixExponential.h
/unsupported/Eigen/src/MatrixFunctions/MatrixFunctionAtomic.h
/unsupported/Eigen/src/MoreVectorization/MathFunctions.h
/unsupported/Eigen/src/LevenbergMarquardt/LevenbergMarquardt.h
/unsupported/Eigen/src/FFT/ei_fftw_impl.h
/unsupported/Eigen/src/FFT/ei_kissfft_impl.h
/unsupported/Eigen/src/Polynomials/PolynomialSolver.h
/unsupported/Eigen/src/Polynomials/Companion.h
/unsupported/Eigen/src/Polynomials/PolynomialUtils.h
/unsupported/Eigen/src/NumericalDiff/NumericalDiff.h
/unsupported/Eigen/src/Skyline/SkylineProduct.h
/unsupported/Eigen/src/Skyline/SkylineMatrixBase.h
/unsupported/Eigen/src/Skyline/SkylineStorage.h
/unsupported/Eigen/src/Skyline/SkylineUtil.h
/unsupported/Eigen/src/Skyline/SkylineInplaceLU.h
/unsupported/Eigen/src/Skyline/SkylineMatrix.h
/unsupported/Eigen/SparseExtra
/unsupported/Eigen/AdolcForward
/unsupported/Eigen/KroneckerProduct
/unsupported/Eigen/NonLinearOptimization
/unsupported/Eigen/BVH
/Eigen/src/misc/Image.h
/Eigen/src/SparseCore/SparseColEtree.h
/Eigen/src/SparseCore/SparseTranspose.h
/Eigen/src/SparseCore/SparseUtil.h
/Eigen/src/SparseCore/SparseCwiseBinaryOp.h
/Eigen/src/SparseCore/SparseDiagonalProduct.h
/Eigen/src/SparseCore/SparseProduct.h
/Eigen/src/SparseCore/SparseDot.h
/Eigen/src/SparseCore/SparseCwiseUnaryOp.h
/Eigen/src/SparseCore/SparseSparseProductWithPruning.h
/Eigen/src/SparseCore/SparseBlock.h
/Eigen/src/SparseCore/SparseDenseProduct.h
/Eigen/src/SparseCore/CompressedStorage.h
/Eigen/src/SparseCore/SparseMatrixBase.h
/Eigen/src/SparseCore/MappedSparseMatrix.h
/Eigen/src/SparseCore/SparseTriangularView.h
/Eigen/src/SparseCore/SparseView.h
/Eigen/src/SparseCore/SparseFuzzy.h
/Eigen/src/SparseCore/TriangularSolver.h
/Eigen/src/SparseCore/SparseSelfAdjointView.h
/Eigen/src/SparseCore/SparseMatrix.h
/Eigen/src/SparseCore/SparseVector.h
/Eigen/src/SparseCore/AmbiVector.h
/Eigen/src/SparseCore/ConservativeSparseSparseProduct.h
/Eigen/src/SparseCore/SparseRedux.h
/Eigen/src/SparseCore/SparsePermutation.h
/Eigen/src/Eigenvalues/RealSchur.h
/Eigen/src/Eigenvalues/ComplexEigenSolver.h
/Eigen/src/Eigenvalues/GeneralizedEigenSolver.h
/Eigen/src/Eigenvalues/ComplexSchur.h
/Eigen/src/Eigenvalues/RealQZ.h
/Eigen/src/Eigenvalues/GeneralizedSelfAdjointEigenSolver.h
/Eigen/src/Eigenvalues/MatrixBaseEigenvalues.h
/Eigen/src/SuperLUSupport/SuperLUSupport.h
/Eigen/src/StlSupport/StdDeque.h
/Eigen/src/StlSupport/StdVector.h
/Eigen/src/StlSupport/StdList.h
/Eigen/src/StlSupport/details.h
/Eigen/src/SparseQR/SparseQR.h
/Eigen/src/LU/Inverse.h
/Eigen/src/LU/arch/Inverse_SSE.h
/Eigen/src/LU/Determinant.h
/Eigen/src/LU/PartialPivLU.h
/Eigen/src/LU/FullPivLU.h
/Eigen/src/Core/BooleanRedux.h
/Eigen/src/Core/Replicate.h
/Eigen/src/Core/arch/AltiVec/PacketMath.h
/Eigen/src/Core/arch/AltiVec/Complex.h
/Eigen/src/Core/arch/SSE/PacketMath.h
/Eigen/src/Core/arch/SSE/Complex.h
/Eigen/src/Core/arch/SSE/MathFunctions.h
/Eigen/src/Core/arch/NEON/PacketMath.h
/Eigen/src/Core/arch/NEON/Complex.h
/Eigen/src/Core/arch/Default/Settings.h
/Eigen/src/Core/CwiseUnaryView.h
/Eigen/src/Core/Array.h
/Eigen/src/Core/ArrayWrapper.h
/Eigen/src/Core/Swap.h
/Eigen/src/Core/Transpositions.h
/Eigen/src/Core/Random.h
/Eigen/src/Core/IO.h
/Eigen/src/Core/SelfCwiseBinaryOp.h
/Eigen/src/Core/VectorwiseOp.h
/Eigen/src/Core/Select.h
/Eigen/src/Core/ArrayBase.h
/Eigen/src/Core/DenseCoeffsBase.h
/Eigen/src/Core/DiagonalProduct.h
/Eigen/src/Core/Assign.h
/Eigen/src/Core/Redux.h
/Eigen/src/Core/ForceAlignedAccess.h
/Eigen/src/Core/BandMatrix.h
/Eigen/src/Core/PlainObjectBase.h
/Eigen/src/Core/DenseBase.h
/Eigen/src/Core/Flagged.h
/Eigen/src/Core/CwiseBinaryOp.h
Mozilla Public License Version 2.0
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

1. Definitions
--------------

1.1. "Contributor"

means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
Licensable by such Contributor to use, reproduce, make available,
modify, display, perform, distribute, and otherwise exploit its
Contributions, either on an unmodified basis, with Modifications, or
as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
for sale, have made, import, and otherwise transfer either its
Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

Open Source Used In DNAC Platform Arica 5642
The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------
3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms
You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.
5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.
*                                                                      *
************************************************************************
************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.
*                                                                      *
************************************************************************

8. Litigation
Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
---------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
--------------------------------------------------------

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

Following applies to:
./doc/UsingIntelMKL.dox
./doc/UsingIntelMKL.dox
./Eigen/src/Eigenvalues/ComplexSchur_MKL.h
./Eigen/src/Eigenvalues/ComplexSchur_MKL.h
./Eigen/src/Eigenvalues/SelfAdjointEigenSolver_MKL.h
./Eigen/src/Eigenvalues/SelfAdjointEigenSolver_MKL.h
./Eigen/src/Eigenvalues/RealSchur_MKL.h
./Eigen/src/Eigenvalues/RealSchur_MKL.h
./Eigen/src/LU/arch/Inverse_SSE.h
./Eigen/src/LU/arch/Inverse_SSE.h
./Eigen/src/LU/PartialPivLU_MKL.h
./Eigen/src/LU/PartialPivLU_MKL.h
./Eigen/src/QR/HouseholderQR_MKL.h
./Eigen/src/QR/HouseholderQR_MKL.h
./Eigen/src/QR/ColPivHouseholderQR_MKL.h
./Eigen/src/QR/ColPivHouseholderQR_MKL.h
./Eigen/src/SVD/JacobiSVD_MKL.h
./Eigen/src/SVD/JacobiSVD_MKL.h
./Eigen/src/PardisoSupport/PardisoSupport.h
./Eigen/src/PardisoSupport/PardisoSupport.h
./Eigen/src/Core/Assign_MKL.h
./Eigen/src/Core/Assign_MKL.h
./Eigen/src/Core/products/SelfadjointMatrixVector_MKL.h

Open Source Used In DNAC Platform Arica 5648
Copyright (c) 2011, Intel Corporation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. *
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. * Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Following applies to:
everything under ./bench/btl

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.
For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.
To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.
When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.
A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include
a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the
violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.
A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.
A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published
by the Free Software Foundation.

If the Program specifies that a proxy can decide which future
versions of the GNU General Public License can be used, that proxy's
public statement of acceptance of a version permanently authorizes you
to choose that version for the Program.

Later license versions may give you additional or different
permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a
later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT
WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND
PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE
DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR
CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES
AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER
OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.
If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type `show w`. This is free software, and you are welcome to redistribute it under certain conditions; type `show c` for details.
The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

------------------------------------------------------------------------
Following applies to:
./test/metis_support.cpp
./test/sparselu.cpp
./unsupported/test/mpreal/mpreal.h
./unsupported/Eigen/src/IterativeSolvers/IterationController.h
./unsupported/Eigen/src/IterativeSolvers/ConstrainedConjGrad.h
./unsupported/Eigen/src/Eigenvalues/ArpackSelfAdjointEigenSolver.h
./Eigen/src/OrderingMethods/Amd.h
./Eigen/src/SparseCholesky/SimplicialCholesky_impl.h

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this
license document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this
license document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of
this License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Following applies to:
./unsupported/Eigen/src/LevenbergMarquardt/LevenbergMarquardt.h
./unsupported/Eigen/src/LevenbergMarquardt/LMcovar.h
./unsupported/Eigen/src/LevenbergMarquardt/LMonestep.h
./unsupported/Eigen/src/LevenbergMarquardt/LMpar.h
./unsupported/Eigen/src/LevenbergMarquardt/LMqrsolv.h

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND, THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY
OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

******************************
freetype2
******************************

FreeType

Quoth http://freetype.sourceforge.net/license.html:

FreeType comes with two licenses from which you can choose the one which fits your needs best.

* The FreeType License is the most commonly used one. It is a BSD-style license with a credit clause (and thus not compatible with the GPL).

* The GNU General Public License (GPL). For all projects which use the GPL also or which need a license compatible to the GPL.

FTL.TXT:
---

The FreeType Project LICENSE
-----------------------------

2006-Jan-27

Copyright 1996-2002, 2006 by David Turner, Robert Wilhelm, and Werner Lemberg
Introduction
===============

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be interested in any kind of bug reports. (`as is' distribution)
- You can use this software for whatever you want, in parts or full form, without having to pay us. (`royalty-free' usage)
- You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. (`credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```
Portions of this software are copyright (C) <year> The FreeType Project (www.freetype.org). All rights reserved.
```

Please replace <year> with the value from the FreeType version you actually use.
Legal Terms
=============

0. Definitions
------------

Throughout this license, the terms 'package', 'FreeType Project', and 'FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the 'FreeType Project', be they named as alpha, beta or final release.

'You' refers to the licensee, or person using the project, where 'using' is a generic term including compiling the project's source code as well as linking it to form a 'program' or 'executable'. This program is referred to as 'a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty
-------------

THE FREETYPE PROJECT IS PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution
-----------------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code
forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

o Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

o Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising
--------------

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts
-----------

There are two mailing lists related to FreeType:

o freetype@nongnu.org
Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution.

If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

o freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

http://www.freetype.org

--- end of FTL.TXT ---

*************************
GL
*************************

Mesa Component Licenses

<table>
<thead>
<tr>
<th>Component</th>
<th>Location</th>
<th>Primary Author</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Mesa code</td>
<td>src/mesa/</td>
<td>Brian Paul</td>
<td>Mesa (MIT)</td>
</tr>
<tr>
<td>Device drivers</td>
<td>src/mesa/drivers/*</td>
<td>See drivers</td>
<td>See drivers</td>
</tr>
<tr>
<td>Ext headers</td>
<td>include/GL/glext.h</td>
<td>SGI</td>
<td>SGI Free B</td>
</tr>
<tr>
<td></td>
<td>include/GL/glxext.h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLUT</td>
<td>src/glut/</td>
<td>Mark Kilgard</td>
<td>Mark's copyright</td>
</tr>
<tr>
<td>GLEW</td>
<td>src/glew-1.13.0</td>
<td>Nigel Stewart</td>
<td>Modified BSD</td>
</tr>
<tr>
<td>Mesa GLU library</td>
<td>src/glu/mesa/</td>
<td>Brian Paul</td>
<td>GNU-LGPL</td>
</tr>
<tr>
<td>SGI GLU library</td>
<td>src/glu/sgi/</td>
<td>SGI</td>
<td>SGI Free B</td>
</tr>
<tr>
<td>demo programs</td>
<td>progs/demos/</td>
<td>various</td>
<td>see source files</td>
</tr>
<tr>
<td>X demos</td>
<td>progs/xdemos/</td>
<td>Brian Paul</td>
<td>see source files</td>
</tr>
<tr>
<td>SGI demos</td>
<td>progs/samples/</td>
<td>SGI</td>
<td>SGI copyright</td>
</tr>
<tr>
<td>RedBook demos</td>
<td>progs/redbook/</td>
<td>SGI</td>
<td>SGI copyright</td>
</tr>
</tbody>
</table>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Alan Hourihane not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Alan Hourihane makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

ALAN HOURIHANE DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL ALAN HOURIHANE BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

---------------------

The OpenGL Extension Wrangler Library
Copyright (C) 2002-2008, Milan Ikits <milan ikits[\]ieee org>
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The name of the author may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

------------------------------------------------------------------------------
License for the slf4j package
------------------------------------------------------------------------------

SLF4J License
Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

These terms are identical to those of the MIT License, also called the X License or the X11 License, which is a simple, permissive non-copyleft free software license. It is deemed compatible with virtually all types of licenses, commercial or otherwise. In particular, the Free Software Foundation has declared it compatible with GNU GPL. It is also known to be approved by the Apache Software Foundation as compatible with Apache Software License.

-------------------------------------------------------------------------------
License for the JUnit package
-------------------------------------------------------------------------------

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program:
where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are not
derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to
reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such
derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other combinations
which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to
Recipient for claims brought by any other entity based on infringement of
intellectual property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby assumes sole
responsibility to secure any other intellectual property rights needed, if any.
For example, if a third party patent license is required to allow Recipient to
distribute the Program, it is Recipient's responsibility to acquire that license
before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

License for the JCIFS package

........................................................................................................

JCIFS License

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.
For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

ICU

There are two licenses here:
- ICU license
- Unicode Terms of Use

ICU License - ICU 1.8.1 and later
From http://source.icu-project.org/repos/icu/icu/trunk/license.html
X License (old version). For license pedigree see the ICU FAQ at http://icu-project.org/usrguide/icufaq.html
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2006 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Unicode Terms of Use, from http://www.unicode.org/copyright.html

For the general privacy policy governing access to this site, see the Unicode Privacy Policy. For trademark usage, see the Unicode Consortium Trademarks and Logo Policy.

Carefully read the following legal agreement ("Agreement"). Use or copying of the software and/or codes provided with this agreement (The "Software") constituting your acceptance of these terms

1. Unicode Copyright.
   1. Copyright 1991-2007 Unicode, Inc. All rights reserved.
   2. Certain documents and files on this website contain a legend indicating that "Modification is permitted." Any person is hereby authorized, without fee, to modify such documents and files to create derivative works conforms to the Unicode Standard, subject to Terms and Conditions herein.
   3. Any person is hereby authorized, without fee, to view, use,
reproduce, and distribute all documents and files solely for informational purposes in the creation of products supporting the Unicode Standard, subject to the Terms and Conditions herein.

4. Further specifications of rights and restrictions pertaining to the use of the particular set of data files known as the "Unicode Character Database" can be found in Exhibit 1.

5. Each version of the Unicode Standard has further specifications of rights and restrictions of use. For the book editions, these are found on the back of the title page. For the online edition, certain files (such as the PDF files for book chapters and code charts) carry specific restrictions. All other files are covered under these general Terms of Use. To request a permission to reproduce any part of the Unicode Standard, please contact the Unicode Consortium.

6. No license is granted to "mirror" the Unicode website where a fee is charged for access to the "mirror" site.

7. Modification is not permitted with respect to this document. All copies of this document must be verbatim.

2. Restricted Rights Legend. Any technical data or software which is licensed to the United States of America, its agencies and/or instrumentalities under this Agreement is commercial technical data or commercial computer software developed exclusively at private expense as defined in FAR 2.101, or DFARS 252.227-7014 (June 1995), as applicable. For technical data, use, duplication, or disclosure by the Government is subject to restrictions as set forth in DFARS 202.227-7015 Technical Data, Commercial and Items (Nov 1995) and this Agreement. For Software, in accordance with FAR 12-212 or DFARS 227-7202, as applicable, use, duplication or disclosure by the Government is subject to the restrictions set forth in this Agreement.

3. Warranties and Disclaimers.

1. This publication and/or website may include technical or typographical errors or other inaccuracies. Changes are periodically added to the information herein; these changes will be incorporated in new editions of the publication and/or website. Unicode may make improvements and/or changes in the product(s) and/or program(s) described in this publication and/or website at any time.

2. If this file has been purchased on magnetic or optical media from Unicode, Inc. the sole and exclusive remedy for any claim will be exchange of the defective media within ninety (90) days of original purchase.

3. EXCEPT AS PROVIDED IN SECTION C.2, THIS PUBLICATION AND/OR SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. UNICODE AND ITS LICENSORS ASSUME NO RESPONSIBILITY FOR ERRORS OR OMISSIONS IN THIS PUBLICATION AND/OR SOFTWARE OR OTHER DOCUMENTS WHICH ARE REFERENCED BY OR LINKED TO THIS PUBLICATION OR THE UNICODE WEBSITE.

4. Waiver of Damages. In no event shall Unicode or its licensors be liable for any special, incidental, indirect or consequential damages of any kind, or any damages whatsoever, whether or not Unicode was advised of the possibility of the damage, including, without limitation, those resulting from
the following: loss of use, data or profits, in connection with the use, modification or distribution of this information or its derivatives.

5. Trademarks.
   1. Unicode and the Unicode logo are registered trademarks of Unicode, Inc.
   2. This site contains product names and corporate names of other companies. All product names and company names and logos mentioned herein are the trademarks or registered trademarks of their respective owners. Other products and corporate names mentioned herein which are trademarks of a third party are used only for explanation and for the owners' benefit and with no intent to infringe.
   3. Use of third party products or information referred to herein is at the user's risk.

6. Miscellaneous.
   1. Jurisdiction and Venue. This server is operated from a location in the State of California, United States of America. Unicode makes no representation that the materials are appropriate for use in other locations. If you access this server from other locations, you are responsible for compliance with local laws. This Agreement, all use of this site and any claims and damages resulting from use of this site are governed solely by the laws of the State of California without regard to any principles which would apply the laws of a different jurisdiction. The user agrees that any disputes regarding this site shall be resolved solely in the courts located in Santa Clara County, California. The user agrees said courts have personal jurisdiction and agree to waive any right to transfer the dispute to any other forum.
   2. Modification by Unicode Unicode shall have the right to modify this Agreement at any time by posting it to this site. The user may not assign any part of this Agreement without Unicode's prior written consent.
   3. Taxes. The user agrees to pay any taxes arising from access to this website or use of the information herein, except for those based on Unicode's net income.
   4. Severability. If any provision of this Agreement is declared invalid or unenforceable, the remaining provisions of this Agreement shall remain in effect.
   5. Entire Agreement. This Agreement constitutes the entire agreement between the parties.

EXHIBIT 1
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2007 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

Unicode and the Unicode logo are trademarks of Unicode, Inc., and may be registered in some jurisdictions. All other trademarks and registered trademarks mentioned herein are the property of their respective owners.

**********************************

icu
There are two licenses here:
- ICU license
- Unicode Terms of Use

ICU License - ICU 1.8.1 and later
From http://source.icu-project.org/repos/icu/icu/trunk/license.html
X License (old version). For license pedigree see the
ICU FAQ at http://icu-project.org/userguide/icufaq.html

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2006 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, and/or sell copies of the
Software, and to permit persons to whom the Software is furnished to do so,
provided that the above copyright notice(s) and this permission notice appear
in all copies of the Software and that both the above copyright notice(s) and
this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY
DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not
be used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Unicode Terms of Use, from http://www.unicode.org/copyright.html

For the general privacy policy governing access to this site, see the
Unicode Privacy Policy. For trademark usage, see the Unicode Consortium
Trademarks and Logo Policy.
Notice to End User: Terms of Use

Carefully read the following legal agreement ("Agreement"). Use or copying of the software and/or codes provided with this agreement (The "Software") constitutes your acceptance of these terms.

1. Unicode Copyright.
   1. Copyright 1991-2007 Unicode, Inc. All rights reserved.
   2. Certain documents and files on this website contain a legend indicating that "Modification is permitted." Any person is hereby authorized, without fee, to modify such documents and files to create derivative works conforming to the Unicode Standard, subject to Terms and Conditions herein.
   3. Any person is hereby authorized, without fee, to view, use, reproduce, and distribute all documents and files solely for informational purposes in the creation of products supporting the Unicode Standard, subject to the Terms and Conditions herein.
   4. Further specifications of rights and restrictions pertaining to the use of the particular set of data files known as the "Unicode Character Database" can be found in Exhibit 1.
   5. Each version of the Unicode Standard has further specifications of rights and restrictions of use. For the book editions, these are found on the back of the title page. For the online edition, certain files (such as the PDF files for book chapters and code charts) carry specific restrictions. All other files are covered under these general Terms of Use. To request a permission to reproduce any part of the Unicode Standard, please contact the Unicode Consortium.
   6. No license is granted to "mirror" the Unicode website where a fee is charged for access to the "mirror" site.
   7. Modification is not permitted with respect to this document.

All copies of this document must be verbatim.

2. Restricted Rights Legend. Any technical data or software which is licensed to the United States of America, its agencies and/or instrumentalities under this Agreement is commercial technical data or commercial computer software developed exclusively at private expense as defined in FAR 2.101, or DFARS 252.227-7014 (June 1995), as applicable. For technical data, use, duplication, or disclosure by the Government is subject to restrictions as set forth in DFARS 202.227-7015 Technical Data, Commercial and Items (Nov 1995) and this Agreement. For Software, in accordance with FAR 12-212 or DFARS 227-7202, as applicable, use, duplication or disclosure by the Government is subject to the restrictions set forth in this Agreement.

3. Warranties and Disclaimers.
   1. This publication and/or website may include technical or typographical errors or other inaccuracies. Changes are periodically added to the information herein; these changes will be incorporated in new editions of the publication and/or website. Unicode may make improvements and/or changes in the product(s) and/or program(s) described in this publication and/or website at any time.
   2. If this file has been purchased on magnetic or optical media from Unicode, Inc. the sole and exclusive remedy for any claim will be exchange
of the defective media within ninety (90) days of original purchase.

3. EXCEPT AS PROVIDED IN SECTION C.2, THIS PUBLICATION AND/OR SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. UNICODE AND ITS LICENSORS ASSUME NO RESPONSIBILITY FOR ERRORS OR OMISSIONS IN THIS PUBLICATION AND/OR SOFTWARE OR OTHER DOCUMENTS WHICH ARE REFERENCED BY OR LINKED TO THIS PUBLICATION OR THE UNICODE WEBSITE.

4. Waiver of Damages. In no event shall Unicode or its licensors be liable for any special, incidental, indirect or consequential damages of any kind, or any damages whatsoever, whether or not Unicode was advised of the possibility of the damage, including, without limitation, those resulting from the following: loss of use, data or profits, in connection with the use, modification or distribution of this information or its derivatives.

5. Trademarks.
   1. Unicode and the Unicode logo are registered trademarks of Unicode, Inc.
   2. This site contains product names and corporate names of other companies. All product names and company names and logos mentioned herein are the trademarks or registered trademarks of their respective owners. Other products and corporate names mentioned herein which are trademarks of a third party are used only for explanation and for the owners' benefit and with no intent to infringe.

5. Use of third party products or information referred to herein is at the user's risk.

6. Miscellaneous.
   1. Jurisdiction and Venue. This server is operated from a location in the State of California, United States of America. Unicode makes no representation that the materials are appropriate for use in other locations. If you access this server from other locations, you are responsible for compliance with local laws. This Agreement, all use of this site and any claims and damages resulting from use of this site are governed solely by the laws of the State of California without regard to any principles which would apply the laws of a different jurisdiction. The user agrees that any disputes regarding this site shall be resolved solely in the courts located in Santa Clara County, California. The user agrees said courts have personal jurisdiction and agree to waive any right to transfer the dispute to any other forum.

   2. Modification by Unicode Unicode shall have the right to modify this Agreement at any time by posting it to this site. The user may not assign any part of this Agreement without Unicode's prior written consent.

   3. Taxes. The user agrees to pay any taxes arising from access to this website or use of the information herein, except for those based on Unicode's net income.

   4. Severability. If any provision of this Agreement is declared invalid or unenforceable, the remaining provisions of this Agreement shall remain in effect.

   5. Entire Agreement. This Agreement constitutes the entire
agreement between the parties.

EXHIBIT 1
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories
http://www.unicode.org/Public/, http://www.unicode.org/reports/, and
http://www.unicode.org/cldr/data/. Unicode Software includes any source code
published in the Unicode Standard or under the directories
http://www.unicode.org/Public/, http://www.unicode.org/reports/, and
http://www.unicode.org/cldr/data/.

NOTICE TO USER: Carefully read the following legal agreement. BY
DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES
("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND
AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF
YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA
FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2007 Unicode, Inc. All rights reserved. Distributed under
the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a
copy of the Unicode data files and any associated documentation (the "Data
Files") or Unicode software and any associated documentation (the "Software")
to deal in the Data Files or Software without restriction, including without
limitation the rights to use, copy, modify, merge, publish, distribute, and/or
sell copies of the Data Files or Software, and to permit persons to whom the
Data Files or Software are furnished to do so, provided that (a) the above
copyright notice(s) and this permission notice appear with all copies of the
Data Files or Software, (b) both the above copyright notice(s) and this
permission notice appear in associated documentation, and (c) there is clear
notice in each modified Data File or in the Software as well as in the
documentation associated with the Data File(s) or Software that the data or
software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD
PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN
THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
 WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING
OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR
SOFTWARE.
Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

Unicode and the Unicode logo are trademarks of Unicode, Inc., and may be registered in some jurisdictions. All other trademarks and registered trademarks mentioned herein are the property of their respective owners.

****************************
java/android_libs/exoplayer
****************************

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

Open Source Used In DNAC Platform Arica 5717
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

******************************************************************************
java/android_libs/protobuf_nano
******************************************************************************

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

******************************
javascript/jquery_ui
******************************
The MIT License (MIT)

Copyright (c) 2015 jQuery Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

******************************
javascript/jquery/v2_0_1
******************************
Copyright 2013 jQuery Foundation and other contributors
http://jquery.com/

https://github.com/jquery/jquery/blob/master/MIT-LICENSE.txt
https://github.com/jquery/sizzle/blob/master/LICENSE
jQuery and Sizzle are released under MIT Licence.

The text is provided below.

MIT License
----

Copyright 2013 jQuery Foundation and other contributors
http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**************************************************
javascript/tracing_framework
**************************************************

Copyright 2012, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

***************************
java_src/android_libs/exoplayer
***************************

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

Open Source Used In DNAC Platform Arica 5723
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*****************************
java_src/android_libs/protobuf_nano/v2
*****************************

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

   * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

************************
jpeg
************************
(extracted from src/README)

LEGAL ISSUES
============

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane. All Rights Reserved except as specified below.
Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA. ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual distribution terms of the Free Software Foundation; principally, that you must include source code if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented
Huffman mode, it is unlikely that very many implementations will support it.)
So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files.
To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that
"The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

libogg

Copyright (c) 2002, Xiph.org Foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
libunwind

Copyright (c) 2002 Hewlett-Packard Co.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libvorbis

Copyright (c) 2002-2008 Xiph.org Foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS `AS IS` AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

******************************************************************************
libxcb
******************************************************************************
Copyright (C) 2001-2006 Bart Massey, Jamey Sharp, and Josh Triplett.
All Rights Reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the
Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute,
sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall
be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors
or their institutions shall not be used in advertising or
otherwise to promote the sale, use or other dealings in this
Software without prior written authorization from the
authors.

******************************************************************************
libxml
Libxml2, an XML C Parser

Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------------------

Copyright (C) 2000,2012 Bjorn Reese and Daniel Veillard.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

Author: breese@users.sourceforge.net

(taken from hash.c)

-----------------------------------------------------------------------------------

Copyright (C) 2000 Gary Pennington and Daniel Veillard.
Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

Author: Gary.Pennington@uk.sun.com

(taken from list.c)

--------------------------------------------------------------------
Copyright (C) 1998 Bjorn Reese and Daniel Stenberg.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

(taken from trio.h and trio.c)

--------------------------------------------------------------------
Copyright (C) 2001 Bjorn Reese <breese@users.sourceforge.net>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

(taken from triodef.h, trionan.h, and trionan.c)

--------------------------------------------------------------------
Copyright (C) 2000 Bjorn Reese and Daniel Stenberg.

Permission to use, copy, modify, and distribute this software for any
permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

(taken from triop.h)

--------------------------------------------------------------------
Copyright (C) 2001 Bjorn Reese and Daniel Stenberg.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

(taken from triostr.h and triostr.c)

*************************************************************************
http://ctrio.sourceforge.net/
*************************************************************************

lodepng

LodePNG

Copyright (c) 2005-2013 Lode Vandevenne

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

******************************
minizip
******************************

******************************
zlib
******************************

(extracted from README, except for match.S)

Copyright notice:

(C) 1995-2004 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

(extracted from match.S, for match.S only)

Copyright (C) 1998, 2007 Brian Raiter <breadbox@muppetlabs.com>

This software is provided 'as-is', without any express or implied warranty. In no event will the author be held liable for any damages.
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

******************************
mongoose
******************************

Copyright (c) 2004-2013 Sergey Lyubka

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

******************************
objective_c/gtm_session_fetcher
******************************

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

*****************************************************
openctm
*****************************************************
Copyright (c) 2009-2010 Marcus Geelnard

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

*****************************************************
OpenCV
*****************************************************
IMPORTANT: READ BEFORE DOWNLOADING, COPYING, INSTALLING OR USING.

By downloading, copying, installing or using the software you agree to this license. 
If you do not agree to this license, do not download, install, 
copy or use the software.

Intel License Agreement
For Open Source Computer Vision Library

Copyright (C) 2000, 2001, Intel Corporation, all rights reserved.
Copyright (C) 2013, OpenCV Foundation, all rights reserved.
Third party copyrights are property of their respective owners.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistribution's of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistribution's in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name of Intel Corporation may not be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the Intel Corporation or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

openssl

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium: https://cla.developers.google.com/clas

Some files from Intel are under yet another license, which is also included underneath.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

/* *

* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.
*/
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. All advertising materials mentioning features or use of this
*    software must display the following acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*    endorse or promote products derived from this software without
*    prior written permission. For written permission, please contact
*    openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*    nor may "OpenSSL" appear in their names without prior written
*    permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*    acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License
-----------------------

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *    "This product includes cryptographic software written by
 *     Eric Young (eay@cryptsoft.com)"
 *    The word 'cryptographic' can be left out if the rouines from the library
 *    being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 *    the apps directory (application code) you must include an acknowledgement:
 *    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

ISC license used for completely new code in BoringSSL:

/* Copyright (c) 2015, Google Inc.
 *
* Permission to use, copy, modify, and/or distribute this software for any
* purpose with or without fee is hereby granted, provided that the above
* copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
* SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
* OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
* CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

Some files from Intel carry the following license:

# Copyright (c) 2012, Intel Corporation
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are
# met:
#
# * Redistributions of source code must retain the above copyright
#   notice, this list of conditions and the following disclaimer.
#
# * Redistributions in binary form must reproduce the above copyright
# This software is provided by Intel Corporation ""as is"" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall Intel Corporation or its contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

***************
openssl/boringssl
***************

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium: https://cla.developers.google.com/clas

Some files from Intel are under yet another license, which is also included underneath.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License
--------------

/* ====================================================================
* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.
* */

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
*
* 3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
* "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following acknowledgment:
* "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*===================================================================
* This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).
Original SSLeay License

/*

Original SSLeay License
-----------------------

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
* 
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscape's SSL.
* 
* This library is free for commercial and non-commercial use as long as
* the following conditions are heared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
* 
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
* must display the following acknowledgement:
* "This product includes cryptographic software written by
* Eric Young (eay@cryptsoft.com)"
* The word 'cryptographic' can be left out if the rouines from the library
* being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
* the apps directory (application code) you must include an acknowledgement:
* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
* 
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

ISC license used for completely new code in BoringSSL:

/* Copyright (c) 2015, Google Inc.
 *
* Permission to use, copy, modify, and/or distribute this software for any
* purpose with or without fee is hereby granted, provided that the above
* copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
* SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
* OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
* CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

Some files from Intel carry the following license:

# Copyright (c) 2012, Intel Corporation
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are
# met:
#
# * Redistributions of source code must retain the above copyright
#   notice, this list of conditions and the following disclaimer.
#
# * Redistributions in binary form must reproduce the above copyright
#   notice, this list of conditions and the following disclaimer in the
#   documentation and/or other materials provided with the
# distribution.
#
# Neither the name of the Intel Corporation nor the names of its
# contributors may be used to endorse or promote products derived from
# this software without specific prior written permission.
#
#
# THIS SOFTWARE IS PROVIDED BY INTEL CORPORATION ""AS IS"" AND ANY
# EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
# PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTEL CORPORATION OR
# CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
# EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
# PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
# PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
# LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
# SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

****************************
pcre
****************************

PCRE is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as
specified below. The documentation for PCRE, supplied in the "doc"
directory, is distributed under the same terms as the software itself. The data
in the testdata directory is not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also
included in the distribution is a set of C++ wrapper functions, and a
just-in-time compiler that can be used to optimize pattern matching. These
are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS

-----------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,
PCRE JUST-IN-TIME COMPILATION SUPPORT
-------------------------------------
Written by:       Zoltan Herczeg
Email local part: hzmester
Email domain:     freemail.hu

Copyright(c) 2010-2015 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER
--------------------------------
Written by:       Zoltan Herczeg
Email local part: hzmester
Email domain:     freemail.hu

Copyright(c) 2009-2015 Zoltan Herczeg
All rights reserved.

THE C++ WRAPPER FUNCTIONS
-------------------------
Contributed by:   Google Inc.

Copyright (c) 2007-2012, Google Inc.
All rights reserved.

THE "BSD" LICENCE
-----------------
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google
  Inc. nor the names of their contributors may be used to endorse or
  promote products derived from this software without specific prior
  written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

End

******************************
pffft
******************************
Copyright (c) 2013 Julien Pommier (pommier@modartt.com)

Based on original fortran 77 code from FFTPACKv4 from NETLIB,
authored by Dr Paul Swarztrauber of NCAR, in 1985.

As confirmed by the NCAR fftpack software curators, the following
FFTPACKv5 license applies to FFTPACKv4 sources. My changes are
released under the same terms.

FFTPACK license:

http://www.cisl.ucar.edu/css/software/ftpack5/ftpk.html

Copyright (c) 2004 the University Corporation for Atmospheric
Research ("UCAR"). All rights reserved. Developed by NCAR's
Computational and Information Systems Laboratory, UCAR,
www.cisl.ucar.edu.

Redistribution and use of the Software in source and binary forms,
with or without modification, is permitted provided that the
following conditions are met:

- Neither the names of NCAR's Computational and Information Systems
Laboratory, the University Corporation for Atmospheric Research, nor the names of its sponsors or contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

- Redistributions of source code must retain the above copyright notices, this list of conditions, and the disclaimer below.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer below in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

***************************************

png

***************************************

libpng

This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

libpng versions 1.2.6, August 15, 2004, through 1.2.27, April 29, 2008, are Copyright (c) 2004, 2006-2008 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors

Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5 - October 3, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6
with the following individuals added to the list of Contributing Authors

Simon-Pierre Cadieux  
Eric S. Raymond  
Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane  
Glenn Randers-Pehrson  
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger  
Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler  
Kevin Bracey  
Sam Bushell  
Magnus Holmgren  
Greg Roelofs  
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger  
Dave Martindale  
Guy Eric Schalnat  
Paul Schmidt  
Tim Wegner
The PNG Reference Library is supplied "AS IS". The Contributing Authors
and Group 42, Inc. disclaim all warranties, expressed or implied,
including, without limitation, the warranties of merchantability and of
fitness for any purpose. The Contributing Authors and Group 42, Inc.
assume no liability for direct, indirect, incidental, special, exemplary,
or consequential damages, which may result from the use of the PNG
Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this
source code, or portions hereof, for any purpose, without fee, subject
to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not
   be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any
   source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without
fee, and encourage the use of this source code as a component to
supporting the PNG file format in commercial products. If you use this
source code in a product, acknowledgment is not required but would be
appreciated.

A "png_get_copyright" function is available, for convenient use in "about"
boxes and the like:

```c
printf("%s",png_get_copyright(NULL));
```

Also, the PNG logo (in PNG format, of course) is supplied in the
files "pngbar.png" and "pngbar.jpg (88x31) and "pngnow.png" (98x31).

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a
certification mark of the Open Source Initiative.

Glenn Randers-Pehrson
glenrp at users.sourceforge.net
April 29, 2008

************************************************************************
protobuf
************************************************************************

Copyright 2008, Google Inc.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

******************************
re2
******************************

// Copyright (c) 2009 The RE2 Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
//  * Neither the name of Google Inc. nor the names of its
//  contributors may be used to endorse or promote products derived from
//  this software without specific prior written permission.
//
//  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
//  "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
//  LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
//  A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
//  OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
//  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
//  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
//  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
//  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
//  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
//  OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*****************************************************************************

stllib
*****************************************************************************

License for STLIB - A collection of public-domain single-file C/C++
libraries, primarily aimed at game developers.

The compilation and test files are licensed under the MIT license, but the
single-file libraries themselves are in the public domain (free for use and
modification for any purpose without legal friction).

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
stl

The STL portion of GNU libstdc++ that is used with gcc3 and gcc4 is licensed under the GPL, with the following exception:

# As a special exception, you may use this file as part of a free software
# library without restriction. Specifically, if other files instantiate
# templates or use macros or inline functions from this file, or you compile
# this file and link it with other files to produce an executable, this
# file does not by itself cause the resulting executable to be covered by
# the GNU General Public License. This exception does not however
# invalidate any other reasons why the executable file might be covered by
# the GNU General Public License.

tinyxml

TinyXml is released under the zlib license:

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

tz
With a few exceptions, all files in the tz code and data (including this one) are in the public domain. The exceptions are tzcode's date.c, newstrftime.3, and strftime.c, which contain material derived from BSD and which use the BSD 3-clause license.

*******************************
utf
******************************

UTF-8 Library

The authors of this software are Rob Pike and Ken Thompson.
Copyright (c) 1998-2002 by Lucent Technologies.
Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software.
THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHORS NOR LUCENT TECHNOLOGIES MAKE ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

*******************************
xmpmeta
*******************************

xmpmeta. A fast XMP metadata parsing and writing library.
Copyright 2016 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

******************************************************************************

Xorg
******************************************************************************
The following is the 'standard copyright' agreed upon by most contributors,
and is currently the canonical license preferred by the X.Org Foundation.
This is a slight variant of the common MIT license form published by the
Open Source Initiative at http://www.opensource.org/licenses/mit-license.php

Copyright holders of new code should use this license statement where
possible, and insert their name to this list. Please sort by surname
for people, and by the full name for other entities (e.g. Juliusz
Chroboczek sorts before Intel Corporation sorts before Daniel Stone).

See each individual source file or directory for the license that applies
to that file.

Copyright (C) 2003-2006,2008 Jamey Sharp, Josh Triplett
Copyright 2009 Red Hat, Inc.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next
paragraph) shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

******************************************************************************
The following licenses are 'legacy' - usually MIT/X11 licenses with the name of the copyright holder(s) in the license statement:

Copyright 1984-1994, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

X Window System is a trademark of The Open Group.

----------------------------------------

Copyright 2000 The XFree86 Project, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.


Portions Copyright 1990, 1991 by Tektronix, Inc.

Permission to use, copy, modify and distribute this documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in all copies, and that the names of Digital and Tektronix not be used in in advertising or publicity pertaining to this documentation without specific, written prior permission. Digital and Tektronix makes no representations about the suitability of this documentation for any purpose. It is provided "as is" without express or implied warranty.

----------------------------------------

Copyright (c) 1999-2000 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE FREE SOFTWARE FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the Free Software Foundation shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the
Free Software Foundation.

----------------------------------------

Code and supporting documentation (c) Copyright 1990 1991 Tektronix, Inc.
All Rights Reserved

This file is a component of an X Window System-specific implementation of Xcms based on the TekColor Color Management System. TekColor is a trademark of Tektronix, Inc. The term "TekHVC" designates a particular color space that is the subject of U.S. Patent No. 4,985,853 (equivalent foreign patents pending). Permission is hereby granted to use, copy, modify, sell, and otherwise distribute this software and its documentation for any purpose and without fee, provided that:

1. This copyright, permission, and disclaimer notice is reproduced in all copies of this software and any modification thereof and in supporting documentation;
2. Any color-handling application which displays TekHVC color coordinates identifies these as TekHVC color coordinates in any interface that displays these coordinates and in any associated documentation;
3. The term "TekHVC" is always used, and is only used, in association with the mathematical derivations of the TekHVC Color Space, including those provided in this file and any equivalent pathways and mathematical derivations, regardless of digital (e.g., floating point or integer) representation.

Tektronix makes no representation about the suitability of this software for any purpose. It is provided "as is" and with all faults.

TEKTRONIX DISCLAIMS ALL WARRANTIES APPLICABLE TO THIS SOFTWARE, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL TEKTRONIX BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA, OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE, OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR THE PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

(c) Copyright 1995 FUJITSU LIMITED
This is source code modified by FUJITSU LIMITED under the Joint Development Agreement for the CDE/Motif PST.

----------------------------------------

Copyright 1992 by Oki Technosystems Laboratory, Inc.
Copyright 1992 by Fuji Xerox Co., Ltd.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Oki Technosystems Laboratory and Fuji Xerox not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

Oki Technosystems Laboratory and Fuji Xerox make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.


Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

FUJITSU LIMITED makes no representations about the suitability of this software for any purpose.

It is provided "as is" without express or implied warranty.

FUJITSU LIMITED DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Copyright (c) 1995 David E. Wexelblat. All rights reserved

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DAVID E. WEXELBLAT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of David E. Wexelblat shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from David E. Wexelblat.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name OMRON not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTUOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Portions Copyright 1990, 1991 by Tektronix, Inc

Rewritten for X.org by Chris Lee <clee@freedesktop.org>

Permission to use, copy, modify, distribute, and sell this documentation for any purpose and without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. Chris Lee makes no representations about the suitability for any purpose of the information in this document. It is provided "as-is" without express or implied warranty.

Copyright 1993 by Digital Equipment Corporation, Maynard, Massachusetts, Copyright 1994 by FUJITSU LIMITED
Copyright 1994 by Sony Corporation

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Digital, FUJITSU LIMITED and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

DIGITAL, FUJITSU LIMITED AND SONY CORPORATION DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL, FUJITSU LIMITED AND SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 1991 by the Open Software Foundation
Software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Silicon Graphics not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission. Silicon Graphics makes no representation about the suitability of this software for any purpose. It is provided "as is" without any express or implied warranty.

Silicon Graphics disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness for a particular purpose. In no event shall Silicon Graphics be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

----------------------------------------

Copyright 1993 by Digital Equipment Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED and Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FUJITSU LIMITED and Digital Equipment Corporation makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

FUJITSU LIMITED AND DIGITAL EQUIPMENT CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU LIMITED AND DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED, Fujitsu Open Systems Solutions, Inc. and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FUJITSU LIMITED, Fujitsu Open Systems Solutions, Inc. and Sony Corporation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

FUJITSU LIMITED, FUJITSU OPEN SYSTEMS SOLUTIONS, INC. AND SONY CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU OPEN SYSTEMS SOLUTIONS, INC., FUJITSU LIMITED AND SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Digital not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

DIGITAL DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.

----------------------------------------
Copyright 1993 by SunSoft, Inc.
Copyright 1999-2000 by Bruno Haible

Permission to use, copy, modify, distribute, and sell this software
and its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appear in all copies and
that both that copyright notice and this permission notice appear
in supporting documentation, and that the names of SunSoft, Inc. and
Bruno Haible not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior
permission. SunSoft, Inc. and Bruno Haible make no representations
about the suitability of this software for any purpose. It is
provided "as is" without express or implied warranty.

SunSoft Inc. AND Bruno Haible DISCLAIM ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS, IN NO EVENT SHALL SunSoft, Inc. OR Bruno Haible BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the TOSHIBA Corp.

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the names of Open Software Foundation and TOSHIBA
not be used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission. Open Software
Foundation and TOSHIBA make no representations about the suitability of this
software for any purpose. It is provided "as is" without express or
implied warranty.

OPEN SOFTWARE FOUNDATION AND TOSHIBA DISCLAIM ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL OPEN SOFTWARE FOUNDATIONN OR TOSHIBA BE
LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1988 by Wyse Technology, Inc., San Jose, Ca.,

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name Wyse not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

WYSE DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1991 by the Open Software Foundation
Copyright 1993, 1994 by the Sony Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Open Software Foundation and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Open Software Foundation and Sony Corporation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OPEN SOFTWARE FOUNDATION AND SONY CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OPEN SOFTWARE FOUNDATION OR SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1992, 1993 by FUJITSU LIMITED
Copyright 1993 by Fujitsu Open Systems Solutions, Inc.

Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED and Fujitsu Open Systems Solutions, Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FUJITSU LIMITED and Fujitsu Open Systems Solutions, Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

FUJITSU LIMITED AND FUJITSU OPEN SYSTEMS SOLUTIONS, INC. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU OPEN SYSTEMS SOLUTIONS, INC. AND FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1993, 1994 by Sony Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Sony Corporation makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

SONY CORPORATION DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1986, 1998  The Open Group
Copyright (c) 2000  The XFree86 Project, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM OR THE XFREE86 PROJECT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium or of the XFree86 Project shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium and the XFree86 Project.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation, and Nippon Telegraph and Telephone Corporation
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of OMRON, NTT Software, NTT, and Open Software Foundation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON, NTT Software, NTT, and Open Software Foundation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or
implied warranty.

OMRON, NTT SOFTWARE, NTT, AND OPEN SOFTWARE FOUNDATION
DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT
SHALL OMRON, NTT SOFTWARE, NTT, OR OPEN SOFTWARE FOUNDATION BE
LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1988 by Wyse Technology, Inc., San Jose, Ca,
Copyright 1987 by Digital Equipment Corporation, Maynard, Massachusetts,

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name Digital not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

DIGITAL AND WYSE DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO
EVENT SHALL DIGITAL OR WYSE BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1991, 1992 by Fuji Xerox Co., Ltd.
Copyright 1992, 1993, 1994 by FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software
and its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appear in all copies and
that both that copyright notice and this permission notice appear
in supporting documentation, and that the name of Fuji Xerox,
FUJITSU LIMITED not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior
permission. Fuji Xerox, FUJITSU LIMITED make no representations
about the suitability of this software for any purpose. 
It is provided "as is" without express or implied warranty.

FUJI XEROX, FUJITSU LIMITED DISCLAIM ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJI XEROX,
FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA
OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 2006 Josh Triplett

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

----------------------------------------

(c) Copyright 1996 by Sebastien Marineau and Holger Veit
<marineau@genie.uottawa.ca>
<Holger.Veit@gmd.de>

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL HOLGER VEIT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Sebastien Marineau or Holger Veit shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from Holger Veit or Sebastien Marineau.

----------------------------------------
Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation, and Nippon Telegraph and Telephone Corporation
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the TOSHIBA Corp.
Copyright 1993, 1994 by Sony Corporation
Copyright 1993, 1994 by the FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of OMRON, NTT Software, NTT, Open Software Foundation, and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON, NTT Software, NTT, Open Software Foundation, and Sony Corporation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON, NTT SOFTWARE, NTT, OPEN SOFTWARE FOUNDATION, AND SONY CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON, NTT SOFTWARE, NTT, OPEN SOFTWARE FOUNDATION, OR SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Bruno Haible not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Bruno Haible makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Bruno Haible DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL Bruno Haible BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright  2003 Keith Packard

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Keith Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Keith Packard makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

KEITH PACKARD DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL KEITH PACKARD BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright (c) 2007-2009, Troy D. Hanson
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----------------------------------------

Copyright 1992, 1993 by TOSHIBA Corp.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of TOSHIBA not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. TOSHIBA make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

TOSHIBA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL TOSHIBA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright IBM Corporation 1993

All Rights Reserved

License to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of IBM not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

IBM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS, IN NO EVENT SHALL IBM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation, and Nippon Telegraph and Telephone Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of OMRON, NTT Software, and NTT not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON, NTT Software, and NTT make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON, NTT SOFTWARE, AND NTT, DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON, NTT SOFTWARE, OR NTT, BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

**************************************************************************

zlib
**************************************************************************

(extracted from README, except for match.S)

Copyright notice:
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes. Please read the FAQ for more information on the distribution of modified source versions.

(extracted from match.S, for match.S only)

Copyright (C) 1998, 2007 Brian Raiter <breadbox@muppetlabs.com>

This software is provided 'as-is', without any express or implied warranty. In no event will the author be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

******************************************************************************
googleurl
******************************************************************************
Copyright 2007, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

******************************************************************************

The file url_parse.cc is based on nsURLParsers.cc from Mozilla. This file is licensed separately as follows:

The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the
License.

The Original Code is mozilla.org code.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by the Initial Developer are Copyright (C) 1998 the Initial Developer. All Rights Reserved.

Contributor(s):
Darin Fisher (original author)

Alternatively, the contents of this file may be used under the terms of either the GNU General Public License Version 2 or later (the "GPL"), or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), in which case the provisions of the GPL or the LGPL are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of either the GPL or the LGPL, and not to allow others to use your version of this file under the terms of the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL or the LGPL. If you do not delete the provisions above, a recipient may use your version of this file under the terms of any one of the MPL, the GPL or the LGPL.

-------------------------------------------------------------------------------

The file icu_utf.cc is from IBM. This file is licensed separately as follows:

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2009 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY
SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

******************************************************************************

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*******************************
jsoncpp
*******************************
The JsonCpp library's source code, including accompanying documentation, tests and demonstration applications, are licensed under the following conditions...

The author (Baptiste Lepilleur) explicitly disclaims copyright in all jurisdictions which recognize such a disclaimer. In such jurisdictions, this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of 2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur, and is released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this software may choose to accept it either as 1) Public Domain, 2) under the conditions of the MIT License (see below), or 3) under the terms of dual Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is
described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

========================================================================
Copyright (c) 2007-2010 Baptiste Lepilleur

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
========================================================================
(END LICENSE TEXT)

The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the Public Domain "license" you can re-license your copy using whatever license you like.

****************************************
libwebp
****************************************
Copyright (c) 2010, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

****************************************
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

QT AUTOMOTIVE SUITE LICENSE AGREEMENT
Agreement version 3.0

This Qt Automotive Suite License Agreement (Agreement) is a legal agreement between The Qt Company (as defined below) and the Licensee (as defined below) for the license of Licensed Software (as defined below). Capitalized terms used herein are defined in Section 1.

WHEREAS:

(A) Licensee wishes to use the Licensed Software for the purpose of developing and distributing Applications and/or Devices; and

(B) The Qt Company is willing to grant the Licensee a right to use Licensed Software for such purpose pursuant to term and conditions of this Agreement.

NOW, THEREFORE, THE PARTIES HEREBY AGREE AS FOLLOWS:

1. DEFINITIONS

"Affiliate" of a Party shall mean an entity (i) which is directly or indirectly controlling such Party; (ii) which is under the same direct or indirect ownership or control as such Party; or (iii) which is directly or indirectly owned or controlled by such Party. For these purposes, an entity shall be treated as being controlled by another if that other entity has fifty percent
(50 %) or more of the votes in such entity, is able to direct its affairs and/or to control the composition of its board of directors or equivalent body.

"Applications" shall mean Licensee's software products created using the Licensed Software in connection with the Program, which may include the Redistributables, or part thereof.

"Contractor(s)" shall mean third party consultants, distributors and contractors performing services to a Party under applicable contractual arrangement.

"Customer(s)" shall mean Licensee's end users to whom Licensee, directly or indirectly, distributes copies of the Redistributables.

"Deployment Platforms" shall mean operating systems specified in the License Certificate, in which the Redistributables can be distributed pursuant to the terms and conditions of this Agreement.

"Designated User(s)" shall mean the employee(s) of Licensee or Licensee's Affiliates acting within the scope of their employment or Licensee's Contractors acting within the scope of their services for Licensee and on behalf of Licensee. Designated Users shall be named in the License Certificate.

"Development License" shall mean the license needed by the Licensee for each Designated User to use the Licensed Software under the license grant described in Section 3.1 of this Agreement.

"Development Platforms" shall mean those operating systems specified in the License Certificate, in which the Licensed Software can be used under the Development License, but not distributed in any form or used for any other purpose.

"Devices" shall mean hardware devices or products that 1) are manufactured and/or distributed by the Licensee or its Affiliates or Contractors in connection with the Program, and (2)(i) incorporate or integrate the Redistributables or parts thereof; or (ii) do not incorporate or integrate the Redistributables at the time of distribution, but where, when used by a Customer, the main user interface or substantial functionality of such device is provided by Application(s) or otherwise depends on the Licensed Software.

"Distribution License(s)" shall mean the license required for distribution of Redistributables in accordance with the license grant described in Section 3.2(ii)-(iii) of this Agreement.

"Distribution License Packs" shall mean set of prepaid Distribution Licenses for distribution of Redistributables, as defined in The Qt Company's standard price list, quote, Purchase Order confirmation or in an appendix hereto, as the case may be.
"Initial Support Term" shall mean a time period of twelve (12) months, calculated from the effective date of this Agreement.

"Intellectual Property Rights" shall mean patents (including utility models), design patents, and designs (whether or not capable of registration), chip topography rights and other like protection, copyrights, trademarks, service marks, trade names, logos or other words or symbols and any other form of statutory protection of any kind and applications for any of the foregoing as well as any trade secrets.

"Licensee" shall mean the individual or legal entity that is party to this Agreement, as identified on the signature page hereof.

"License Certificate" shall mean a certificate accompanying the Licensed Software and generated for each Designated User respectively. License Certificate will specify the Designated User, the Development Platforms, Deployment Platforms, Program and the Term of this Agreement. The terms of the License Certificate are considered part of this Agreement and shall be updated from time to time to reflect any changes to the foregoing terms relating to Licensee's rights to the Licensed Software.

"Licensee's Records" shall mean books and records that are likely to contain information bearing on Licensee's compliance with this Agreement or the payments due to The Qt Company under this Agreement, including, but not limited to: assembly logs, sales records and distribution records.

"Licensees SDK Contractors" shall mean Contractors of Licensee, who have purchased or received SDK from the Licensee relating to the Program.

"License Fee" shall mean the fee charged to the Licensee for rights granted under the terms of this Agreement.

"Licensed Software" shall mean all versions of The Qt Company's computer software products, online or electronic documentation, associated media and printed materials, including the source code, example programs and the documentation, licensed to the Licensee under this Agreement. Licensed Software does not include Third Party Software (as defined in Section 4) or Open Source Qt.

"Modified Software" shall mean bug-fixes, error corrections, patches or modifications made to the Licensed Software by Licensee, including documentation related thereto.

"Online Services" shall mean any services or access to systems made available by The Qt Company to the Licensee over the Internet relating to the Licensed Software or for the purpose of use by the Licensee of the Licensed Software or Support. Use of any such Online Services is discretionary for the Licensee and some of them may be subject to additional fees.
"Open Source Qt" shall mean all versions of The Qt Company’s Qt computer software products, online or electronic documentation, associated media and printed materials, including the source code, example programs and the documentation available under the terms of the GNU Lesser General Public License, version 2.1 or later ("LGPL") or the GNU General Public License, version 2.0 or later ("GPL").

"Party" or "Parties" shall mean Licensee and/or The Qt Company.

"Program" shall mean Licensee’s business program for which purpose the Licensee is entitled to use the Licensed Software and grant the Licensee's SDK Contractors a right to use the Licensed Software as part of a SDK.

"Redistributables" shall mean the portions of the Licensed Software set forth in Appendix 1, Section 1 that may be distributed pursuant to the terms of this Agreement in object code form only, including any relevant documentation. Where relevant, any reference to Licensed Software in this Agreement shall include and refer also to Redistributables.

"SDK" or "Software Development Kit" shall mean a combination of software modules including Licensed Software intended to be utilized in connection with the Program.

"Submitted Modified Software" shall have the meaning as set forth in Section 2.3.

"Support" shall mean standard developer support that is provided by The Qt Company to assist Designated Users in using the Licensed Software in accordance with The Qt Company's standard support terms.

"Support Renewal Term" shall mean a time period of twelve (12) months, calculated from the end of the Initial Support Term or previous Support Renewal Term, as applicable.

"Support Term" shall mean the Initial Support Term and any possible Support Renewal Terms(s) during which time the Licensee is eligible to receive for Support for the Licensed Software.

"Taxes" shall have the meaning set forth in Section 10.5.

"Term" shall mean the validity period of this Agreement, as set forth in the License Certificate.

The Qt Company shall mean:

(i) in the event Licensee is an individual residing in the United States or a legal entity incorporated in the United States or having its headquarters in the
United States, The Qt Company Inc., a Delaware corporation with its office at 2350 Mission College Blvd., Suite 1020, Santa Clara, CA 95054, USA.; or

(ii) in the event the Licensee is an individual residing outside of the United States or a legal entity incorporated outside of the United States or having its registered office outside of the United States, The Qt Company Ltd., a Finnish company with its registered office at Bertel Jungin aukio D3A, 02600 Espoo, Finland.

"Updates" shall mean a release or version of the Licensed Software containing bug fixes, error corrections and other changes that are generally made available to users of the Licensed Software that have contracted for Support. Updates are generally depicted as a change to the digits following the decimal in the Licensed Software version number. The Qt Company shall make Updates available to the Licensee under the Support. Updates shall be considered as part of the Licensed Software hereunder.

"Upgrades" shall mean a release or version of the Licensed Software containing enhancements and new features and are generally depicted as a change to the first digit of the Licensed Software version number. In the event Upgrades are provided to the Licensee under this Agreement, they shall be considered as part of the Licensed Software hereunder.

2. OWNERSHIP 2.1

Ownership of The Qt Company

The Licensed Software is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Licensed Software is licensed, not sold.

All The Qt Company's Intellectual Property Rights are and shall remain the exclusive property of The Qt Company or its licensors respectively.

2.2 Ownership of Licensee

All the Licensee's Intellectual Property Rights are and shall remain the exclusive property of the Licensee or its licensors respectively.

All Intellectual Property Rights to the Modified Software, Applications and Devices shall remain with the Licensee and no rights thereto shall be granted by the Licensee to The Qt Company under this Agreement (except as set forth in Section 2.3 below).

2.3 Modified Software

Licensee may create Modified Software that breaks the source or binary compatibility with the Licensed Software. This includes, but is not limited to,
changing the application programming interfaces ("API") by adding, changing or deleting any variable, method, or class signature in the Licensed Software and/or any inter-process protocols, services or standards in the Licensed Software libraries. To the extent that Licensee breaks source or binary compatibility with the Licensed Software, Licensee acknowledges that The Qt Company's ability to provide Support may be prevented or limited and Licensee's ability to make use of Updates may be restricted.

To the extent Licensee submits Modified Software to The Qt Company ("Submitted Modified Software"), Licensee hereby grants The Qt Company a sublicensable, assignable, irrevocable, perpetual, worldwide, non-exclusive, royalty-free and fully paid-up license, under all of Licensee's Intellectual Property Rights, to reproduce, adapt, translate, modify, and prepare derivative works of, publicly display, publicly perform, sublicense, make available and distribute suchSubmitted Modified Software as The Qt Company sees fit at its free and absolute discretion. For the sake of clarity, the Licensee shall have no obligation to provide Modified Software to The Qt Company.

3. LICENSES GRANTED

3.1 Development with Licensed Software

Subject to the terms of this Agreement, The Qt Company grants to Licensee a personal, worldwide, non-exclusive, non-transferable license, valid for the Term, to use, modify and copy the Licensed Software by Designated Users on the Development Platforms for the sole purposes of designing, developing, demonstrating and testing Application(s) and/or Devices, and to provide thereto related support and other services to end-user Customers.

Licensee may install copies of the Licensed Software on an unlimited number of computers provided that (i) only the Designated Users may use the Licensed Software, and (ii) all Designated Users must have a valid Development License to use Licensed Software.

Licensee may at any time designate another Designated User to replace a then-current Designated User by notifying The Qt Company in writing, provided that any Designated User may be replaced only once during any six-month period.

3.2 Distribution of Redistributables

Subject to the terms of this Agreement, The Qt Company grants to Licensee a personal, worldwide, non-exclusive, non-transferable license, valid for the Term, to (i) distribute, by itself or through its Contractors, Redistributables as installed, incorporated or integrated into Applications for execution on the Deployment Platforms, and (ii) distribute, by itself or through one or more tiers of Contractors, Redistributables as installed, incorporated or integrated, or intended to be installed, incorporated or integrated into Devices for execution on the Deployment Platforms, and (iii) grant sublicense
Redistributables, as distributed hereunder, for Customers solely for Customer's internal use and to the extent necessary in order for the Customers to use the Applications and/or Devices for their respective intended purposes.

Right to distribute the Redistributables as provided herein is conditional upon the Licensee having purchased and paid the appropriate amount of Development and Distribution Licenses from The Qt Company before distributing any Redistributables to Customers.

For the avoidance of any doubt it is specifically acknowledged and agreed that distribution of Redistributables solely as installed, incorporated or integrated into Applications for execution on the Deployment Platform(s), as specified in (i) of the first paragraph of Section 3.2 above, i.e. with no connection to Devices or intention to use in connection therewith, shall not require a Distribution License.

3.3 SDK License

The Qt Company grants to Licensee a personal, worldwide, non-exclusive, non-transferable license, valid for the Term, to (i) distribute Licensed Software as a part of the SDK to Licensee's SDK Contractors in connection with the Program and (ii) in connection with the Program, by itself or by Licensee's SDK Contractors, combine, incorporate or integrate Licensed Software with, or use Licensed Software for creation of, any software created with or incorporating Open Source Qt, provided, however, that:

(i) the Licensee's SDK Contractors are only entitled to use the Licensed Software as part of SDK and for the sole purpose of developing software for Devices that are distributed under the Program; and

(ii) Licensee's SDK Contractors shall not be entitled to distribute the SDK or any part thereof to any third parties.

For the avoidance of any doubt, the distribution of such software development tools that do not contain Licensed Software shall not be covered by this Agreement.

3.4 Further Requirements

The licenses granted above in this Section 3 by The Qt Company to Licensee are conditional and subject to Licensee's compliance with the following terms:

(i) Licensee shall not remove or alter any copyright, trademark or other proprietary rights notice contained in any portion of the Licensed Software;

(ii) Applications and SDKs must add primary and substantial functionality to the Licensed Software;
(iii) Applications may not pass on functionality which in any way makes it possible for others to create software with the Licensed Software; provided however that Licensee may use the Licensed Software's scripting and QML ("Qt Quick") functionality solely in order to enable scripting, themes and styles that augment the functionality and appearance of the Application(s) without adding primary and substantial functionality to the Application(s);

(iv) Applications and SDKs must not compete with the Licensed Software;

(v) Licensee shall not use The Qt Company's or any of its suppliers' names, logos, or trademarks to market Applications or SDKs, except that Licensee may use "Built with Qt" logo to indicate that Application(s) was developed using the Licensed Software;

(vi) Except as expressly provided in Section 3.3, Licensee shall not distribute, sublicense or disclose source code of Licensed Software to any third party (provided however that Licensee may appoint employee(s) of Contractors as Designated Users to use Licensed Software pursuant to this Agreement);

(vii) Licensee shall not grant the Customers a right to (i) make copies of the Redistributables except when and to the extent required to use the Applications and/or Devices for their intended purpose, (ii) modify the Redistributables or create derivative works thereof, (iii) disassemble or otherwise reverse engineer Redistributables, or (iv) redistribute any copy or portion of the Redistributables to any third party, except as part of the onward sale of the Device on which the Redistributables are installed;

(viii) Except as expressly provided in Section 3.3, Licensee shall not and shall cause that its Affiliates, Contractors and Licensee's SDK Contractors shall not a) in any way, combine, incorporate or integrate Licensed Software with, or use Licensed Software for creation of, any software created with or incorporating Open Source Qt or b) incorporate or integrate Applications into a hardware device or product other than a Device, unless Licensee has received an advance written permission from The Qt Company to do so. Unless specifically otherwise agreed, any and all distribution by the Licensee during the Term of a hardware device or product a) which incorporate or integrate any part of Licensed Software or Open Source Qt; or b) where the main user interface or substantial functionality is provided by software build with Licensed Software or Open Source Qt or otherwise depends on the Licensed Software or Open Source Qt, shall be considered as distribution under this Agreement and dependent on compliance thereof (including but not limited to obligation to pay applicable License Fees for such distribution);

(ix) Licensee shall cause all of its Affiliates and Contractors entitled to make use of the licenses granted under this Agreement, to be contractually bound to comply with the relevant terms of this Agreement and not to use the Licensed Software beyond the terms hereof and for any purposes other than operating within the scope of their services for Licensee. Licensee shall be responsible
for any and all actions and omissions of its Affiliates and Contractors relating
to the Licensed Software and use thereof (including but not limited to payment
of all applicable License Fees);

(x) Except when and to the extent explicitly provided in this Section 3,
Licensee shall not transfer, publish, disclose, display or otherwise make
available the Licensed Software;

(xi) Licensee shall not take any action inconsistent with The Qt Company's
Intellectual Property Rights; and

(xii) Attempt or enlist a third party to conduct or attempt to conduct any of
the above.

Above terms shall not be applicable if and to the extent they conflict with any
mandatory provisions of any applicable laws.

Any use of Licensed Software beyond the provisions of this Agreement is strictly
prohibited and requires an additional license from The Qt Company.

4. THIRD PARTY SOFTWARE

The Licensed Software may provide links to third party libraries or code
(collectively "Third Party Software") to implement various functions. Third
Party Software does not comprise part of the Licensed Software. In some cases,
access to Third Party Software may be included in the Licensed Software. Such
Third Party Software will be listed in the ".../src/3rdparty" source tree
delivered with the Licensed Software or documented in the Licensed Software, as
such may be amended from time to time. Licensee acknowledges that use or
distribution of Third Party Software is in all respects subject to applicable
license terms of applicable third party right holders. 5. PRE-RELEASE CODE

The Licensed Software may contain pre-release code and functionality marked or
otherwise stated as "Technology Preview", "Alpha", "Beta" or similar
designation. Such pre-release code may be present in order to provide
experimental support for new platforms or preliminary versions of one or more
new functionalities. The pre-release code may not be at the level of performance
and compatibility of a final, generally available, product offering of the
Licensed Software. The pre-release parts of the Licensed Software may not
operate correctly, may contain errors and may be substantially modified by The
Qt Company prior to the first commercial product release, if any. The Qt Company
is under no obligation to make pre-release code commercially available, or
provide any Support or Updates relating thereto. The Qt Company assumes no
liability whatsoever regarding any pre-release code, but any use thereof is
exclusively at Licensee’s own risk and expense.

6. LIMITED WARRANTY AND WARRANTY DISCLAIMER
The Qt Company hereby represents and warrants that it has the power and authority to grant the rights and licenses granted to Licensee under this Agreement.

Except as set forth above, the Licensed Software is licensed to Licensee "as is".

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE QT COMPANY ON BEHALF OF ITSELF AND ITS LICENSORS, SUPPLIERS AND AFFILIATES, DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT WITH REGARD TO THE LICENSED SOFTWARE. THE QT COMPANY DOES NOT WARRANT THAT THE LICENSED SOFTWARE WILL SATISFY LICENSEE'S REQUIREMENTS OR THAT IT WILL OPERATE WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED. ALL USE OF AND RELIANCE ON THE LICENSED SOFTWARE IS AT THE SOLE RISK OF AND RESPONSIBILITY OF LICENSEE.

Licensee's exclusive remedy and The Qt Company's entire liability for Licensed Software shall be limited, at The Qt Company's option, to correction of the error, replacement of the Licensed Software or return of the applicable fees paid for the defective Licensed Software for the time period during which the License is not able to utilize the Licensed Software under the terms of this Agreement.

7. INDEMNIFICATION AND LIMITATION OF LIABILITY

7.1 Limitation of Liability

EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, (II) LICENSEE'S DUTY TO PAY ALL APPLICABLE LICENSE FEES AND COMPENSATIONS, AND (III) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY BE LIABLE TO THE OTHER PARTY FOR ANY LOSS OF PROFIT, LOSS OF DATA, LOSS OF BUSINESS OR GOODWILL OR ANY OTHER INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE COST, DAMAGES OR EXPENSE OF ANY KIND, HOWSOEVER ARISING UNDER OR IN CONNECTION WITH THIS AGREEMENT.

EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, (II) LICENSEE'S DUTY TO PAY ALL APPLICABLE LICENSE FEES AND COMPENSATIONS, AND (III) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY'S TOTAL AGGREGATE LIABILITY UNDER THIS AGREEMENT EXCEED THE AGGREGATE LICENSE FEES RECEIVED BY THE QT COMPANY FROM LICENSEE DURING THE PERIOD OF TWELVE (12) MONTHS IMMEDIATELY PRECEDING THE EVENT RESULTING IN SUCH LIABILITY.

THE PROVISIONS OF THIS SECTION 7 ALLOCATE THE RISKS UNDER THIS AGREEMENT BETWEEN THE QT COMPANY AND LICENSEE AND THE PARTIES HAVE RELIED UPON THE LIMITATIONS SET FORTH HEREIN IN DETERMINING WHETHER TO ENTER INTO THIS AGREEMENT.
7.2 Licensees Indemnification

Licensee shall indemnify and hold harmless The Qt Company from and against any claim, injury, judgment, settlement, loss or expense, including attorneys' fees related to: (a) Licensee's misrepresentation in connection with The Qt Company or the Licensed Software or breach of this Agreement, (b) the Application or Device (except where such cause of liability is solely attributable to the Licensed Software).

8. SUPPORT, UPDATES AND ONLINE SERVICES

Licensee will be eligible to receive Support and Updates and to use the Online Services during the Support Term. Unless otherwise decided by The Company at its free and absolute discretion, Upgrades will not be included in the Support but may be available subject to additional fees.

Licenses granted under this Agreement shall include a prepaid Initial Support Term.

Initial Support Term shall be automatically extended to one or more Support Renewal Term(s), unless and until either Party notifies the other Party in writing that it does not wish to continue the Support, such notification to be provided to the other Party no less than ninety (90) days before expiry of the Initial Support Term or respective Support Renewal Term. During any such Support Renewal Term Support shall be available subject to prices and terms agreed between the Parties or, if no advance agreement exists, subject to The Qt Company's standard pricing applicable at the commencement date of any such Support Renewal Term. From time to time The Qt Company may change Support provided within each Support plan; provided that during the respective Initial Support Term or Support Renewal Term (as the case may be), the level of Support provided by The Qt Company may not be reduced without the consent of the Licensee.

Unless otherwise agreed, The Qt Company shall not be responsible for providing any service or support to the Customers.

9. CONFIDENTIALITY

Each Party acknowledges that during the Term of this Agreement each Party may receive information about the other Party's business, business methods, business plans, customers, business relations, technology, and other information, including the terms of this Agreement, that is confidential and of great value to the other Party, and the value of which would be significantly reduced if disclosed to third parties ("Confidential Information"). Accordingly, when a Party (the "Receiving Party") receives Confidential Information from the other Party (the "Disclosing Party"), the Receiving Party shall only disclose such information to employees and Contractors on a need to know basis, and shall cause its employees and employees of its Affiliates to: (i) maintain any and all
Confidential Information in confidence; (ii) not disclose the Confidential Information to a third party without the Disclosing Party's prior written approval; and (iii) not, directly or indirectly, use the Confidential Information for any purpose other than for exercising its rights and fulfilling its responsibilities pursuant to this Agreement. Each Party shall take reasonable measures to protect the Confidential Information of the other Party, which measures shall not be less than the measures taken by such Party to protect its own confidential and proprietary information.

Obligation of confidentiality shall not apply to information that (i) is or becomes generally known to the public through no act or omission of the Receiving Party; (ii) was in the Receiving Party's lawful possession prior to the disclosure hereunder and was not subject to limitations on disclosure or use; (iii) is developed independently by employees or Contractors of the Receiving Party or other persons working for the Receiving Party who have not had access to the Confidential Information of the Disclosing Party, as proven by the written records of the Receiving Party; (iv) is lawfully disclosed to the Receiving Party without restrictions, by a third party not under an obligation of confidentiality; or (v) the Receiving Party is legally compelled to disclose, in which case the Receiving Party shall notify the Disclosing Party of such compelled disclosure and assert the privileged and confidential nature of the information and cooperate fully with the Disclosing Party to limit the scope of disclosure and the dissemination of disclosed Confidential Information to the minimum extent necessary.

The obligations under this Section 9 shall continue to remain in force for a period of five (5) years after the last disclosure, and, with respect to trade secrets, for so long as such trade secrets are protected under applicable trade secret laws.

10. FEES, DELIVERY AND PAYMENT

10.1 License Fees

License Fees are described in The Qt Company's standard price list, quote or Purchase Order confirmation or in an appendix hereto, as the case may be. The License Fees shall not be refunded or claimed as a credit, even on the ground that Distribution Licenses are not used, i.e. Redistributables are not actually distributed corresponding to the Distribution Licenses purchased, or for any other reason.

10.2 Ordering Licenses

Licensee may purchase Development Licenses and Distribution Licenses pursuant to agreed pricing terms or, if no specific pricing terms have been agreed upon, at The Qt Company's standard pricing terms applicable at the time of purchase.

Licensee shall submit all purchase orders for Development Licenses and
Distribution Licenses to The Qt Company by email or any other method acceptable to The Qt Company (each such order is referred to herein as a "Purchase Order") for confirmation, whereupon the Purchase Order shall become binding between the Parties.

10.3 Distribution

License Packs Unless otherwise agreed, the Distribution Licenses are bought by way of Distribution License Packs.

Upon due payment of the ordered Distribution License Pack(s), the Licensee will have an account of Distribution Licenses available for installing, bundling or integrating (all jointly "installing") the Redistributables with the Devices or for otherwise distributing the Redistributables in accordance with this Agreement.

Each time Licensee "installs" or distributes a copy of Redistributables, then one Distribution License is used, and Licensee's account of available Distribution Licenses is decreased accordingly.

Licensee may "install" copies of the Redistributables so long as Licensee has Distribution Licenses remaining on its account.

Redistributables will be deemed to have been "installed" into a Device when one of the following circumstances shall have occurred: a) the Redistributables have been loaded onto the Device and used outside of the Licensee's premises or b) the Device has been fully tested and placed into Licensee's inventory (or sold) for the first time (i.e., Licensee will not be required to use (or pay for) more than one Distribution License for each individual Device, e.g. in a situation where a Device is returned to Licensee's inventory after delivery to a distributor or sale to a Customer). In addition, if Licensee includes a back-up copy of the Redistributables on a CD-ROM or other storage medium along with the product, that backup copy of the Redistributables will not be deemed to have been "installed" and will not require an additional Distribution License.

10.4 Payment Terms

License Fees and any other charges under this Agreement shall be paid by Licensee no later than thirty (30) days from the date of the applicable invoice from The Qt Company.

The Qt Company will submit an invoice to Licensee after the date of this Agreement and/or after The Qt Company receives a Purchase Order from Licensee. A late payment charge of the lower of (a) one percent per month; or (b) the interest rate stipulated by applicable law, shall be charged on any unpaid balances that remain past due.

The Qt Company shall have the right to suspend, terminate or withhold grants of
all rights to the Licensed Software hereunder, including but not limited to the Developer License, Distribution License, and Support, should Licensee fail to make payment in a timely fashion.

10.5 Taxes
All License Fees and other charges payable hereunder are gross amounts but exclusive of any value added tax, use tax, sales tax and other taxes, duties or tariffs ("Taxes"). Such applicable Taxes shall be paid by Licensee, or, where applicable, in lieu of payment of such Taxes, Licensee shall provide an exemption certificate to The Qt Company and any applicable authority.

11 RECORD-KEEPING AND REPORTING OBLIGATIONS; AUDIT RIGHTS

11.1 Licensee's Record-keeping

Licensee shall at all times maintain accurate and up-to-date written records of Licensee's activities related to the use of Licensed Software and distribution of Redistributables. The records shall be adequate to determine Licensee's compliance with the provisions of this Agreement and to demonstrate the number of Designated Users and Redistributables distributed by Licensee. The records shall conform to good accounting practices reasonably acceptable to The Qt Company.

Licensee shall, within thirty (30) days from the end of each calendar quarter, deliver to The Qt Company a report detailing the number of Designated Users and copies of Redistributables distributed by Licensee during that calendar quarter, and also detailing the number of undistributed copies of Redistributables made by Licensee and remaining in its account (i.e., undistributed copies for which Distribution Licenses have been or need to be obtained from The Qt Company). Such report shall contain such other information as The Qt Company shall reasonably require from time to time.

11.2. The Qt Company's Audit Rights

The Qt Company or an independent auditor acting on behalf of The Qt Company's, may, upon at least five (5) business days' prior written notice and at its expense, audit Licensee with respect to the use of the Redistributables, but not more frequently than once during each 6-month period. Such audit may be conducted by mail, electronic means or through an in-person visit to Licensee's place of business. Any such in-person audit shall be conducted during regular business hours at Licensee's facilities and shall not unreasonably interfere with Licensee's business activities. The Qt Company or the independent auditor acting on behalf of The Qt Company shall be entitled to inspect Licensee's Records. All such Licensee's Records and use thereof shall be subject to an obligation of confidentiality under this Agreement.

If an audit reveals that Licensee is using the Licensed Software beyond scope of the licenses Licensee has paid for, Licensee agrees to immediately pay The Qt
Company any amounts owed for such unauthorized use.

In addition, in the event the audit reveals a material violation of the terms of this Agreement (underpayment of more than 5% of License Fees shall always be deemed a material violation for purposes of this section), then the Licensee shall pay The Qt Company's reasonable cost of conducting such audit.

12 TERM AND TERMINATION

12.1 Term

This Agreement shall enter into force upon due acceptance by both Parties and remain in force for the Term, unless and until terminated pursuant to the terms of this Section 12.

12.2 Termination by The Qt Company

The Qt Company shall have the right to terminate this Agreement upon thirty (30) days prior written notice if (i) the Licensee is in material breach of any obligation of this Agreement and fails to remedy such breach within such notice period; (ii) or Licensee or any of its Affiliates bring a suit before any court or administrative agency or otherwise assert a claim against The Qt Company's or any of its Affiliates' Intellectual Property Rights or validity thereof.

12.3 Mutual Right to Terminate

Either Party shall have the right to terminate this Agreement immediately upon written notice in the event that the other Party becomes insolvent, files for any form of bankruptcy, makes any assignment for the benefit of creditors, has a receiver, administrative receiver or officer appointed over the whole or a substantial part of its assets, ceases to conduct business, or an act equivalent to any of the above occurs under the laws of the jurisdiction of the other Party.

12.4 Parties Rights and Duties upon Termination

Upon expiry or termination of the Agreement for any reason, Licensee shall, within 30 days after such termination, cease and shall cause all Designated Users (including those of its Affiliates' and Contractors') and Licensee's SDK Contractors to cease using the Licensed Software and distribution of the Redistributables under this Agreement. Notwithstanding the above, in the event the Agreement expires or is terminated for reason other than by The Qt Company pursuant to Section 12.2, the Licensee is entitled, for a period of six (6) months after the effective date of termination, to continue distribution of Devices under the Distribution Licenses paid but unused at such effective date of termination.

Upon any such termination the Licensee shall destroy or return to The Qt
Company all copies of the Licensed Software and all related materials and will certify the same to The Qt Company upon its request, provided however that Licensee may retain and exploit such copies of the Licensed Software as it may reasonably require in providing continued support to Customers.

Expiry or termination of this Agreement for any reason whatsoever shall not relieve Licensee of its obligation to pay any License Fees accrued or payable to The Qt Company prior to the effective date of termination, and Licensee shall immediately pay to The Qt Company all such fees upon the effective date of termination. Termination of this Agreement shall not affect any rights of Customers to continue use of Applications and Devices (and therein incorporated Redistributables).

13. GOVERNING LAW AND LEGAL VENUE

In the event this Agreement is in the name of The Qt Company Inc., a Delaware Corporation, then:

(i) this Agreement shall be construed and interpreted in accordance with the laws of the State of California, USA, excluding its choice of law provisions;

(ii) the United Nations Convention on Contracts for the International Sale of Goods will not apply to this Agreement; and

(iii) any dispute, claim or controversy arising out of or relating to this Agreement or the breach, termination, enforcement, interpretation or validity thereof, including the determination of the scope or applicability of this Agreement to arbitrate, shall be determined by arbitration in San Francisco, USA, before one arbitrator. The arbitration shall be administered by JAMS pursuant to JAMS' Streamlined Arbitration Rules and Procedures. Judgment on the Award may be entered in any court having jurisdiction. This Section shall not preclude parties from seeking provisional remedies in aid of arbitration from a court of appropriate jurisdiction.

In the event this Agreement is in the name of The Qt Company Ltd., a Finnish Company, then:

(i) this Agreement shall be construed and interpreted in accordance with the laws of Finland, excluding its choice of law provisions;

(ii) the United Nations Convention on Contracts for the International Sale of Goods will not apply to this Agreement; and

(iii) any disputes, controversy or claim arising out of or relating to this Agreement, or the breach, termination or validity thereof shall be finally settled by arbitration in accordance with the Arbitration Rules of Finland Chamber of Commerce. The arbitration tribunal shall consist of one (1), or if either Party so requires, of three (3), arbitrators. The award shall be
final and binding and enforceable in any court of competent jurisdiction. The arbitration shall be held in Helsinki, Finland and the process shall be conducted in the English language. This Section shall not preclude parties from seeking provisional remedies in aid of arbitration from a court of appropriate jurisdiction.

14. GENERAL PROVISIONS

14.1 No Assignment Licensee

shall not be entitled to assign or transfer all or any of its rights, benefits and obligations under this Agreement without the prior written consent of The Qt Company, which shall not be unreasonably withheld or delayed. The Qt Company shall be entitled to freely assign or transfer any of its rights, benefits or obligations under this Agreement.

14.2 No Third Party Representations

Licensee shall make no representations or warranties concerning the Licensed Software on behalf of The Qt Company. Any representation or warranty Licensee makes or purports to make on The Qt Company’s behalf shall be void as to The Qt Company.

14.3 Surviving Sections

Any terms and conditions that by their nature or otherwise reasonably should survive termination of this Agreement shall so be deemed to survive.

14.4 Entire Agreement

This Agreement, the exhibits hereto, the License Certificate and any applicable Purchase Order constitute the complete agreement between the Parties and supersedes all prior or contemporaneous discussions, representations, and proposals, written or oral, with respect to the subject matters discussed herein. In the event of any conflict or inconsistency between this Agreement and any Purchase Order, the terms of this Agreement will prevail over the terms of the Purchase Order with respect to such conflict or inconsistency.

14.5 Modifications

No modification of this Agreement shall be effective unless contained in a writing executed by an authorized representative of each Party. No term or condition contained in Licensee's Purchase Order shall apply unless expressly accepted by The Qt Company in writing.

14.6 Force Majeure

Except for the payment obligations hereunder, neither Party shall be liable to
the other for any delay or non-performance of its obligations hereunder in the event and to the extent that such delay or non-performance is due to an event of act of God, terrorist attack or other similar unforeseeable catastrophic event that prevents either Party for fulfilling its obligations under this Agreement and which such Party cannot avoid or circumvent (“Force Majeure Event”). If the Force Majeure Event results in a delay or non-performance of a Party for a period of three (3) months or longer, then either Party shall have the right to terminate this Agreement with immediate effect without any liability (except for the obligations of payment arising prior to the event of Force Majeure) towards the other Party.

14.7 Notices

Any notice given by one Party to the other shall be deemed properly given and deemed received if specifically acknowledged by the receiving Party in writing or when successfully delivered to the recipient by hand, fax, or special courier during normal business hours on a business day to the addresses specified for The Qt Company in the beginning of this Agreement, and for the Licensee in the Licensees account profile. Each communication and document made or delivered by one Party to the other Party pursuant to this Agreement shall be in the English language.

14.8 Export Control
Licensee acknowledges that the Redistributables may be subject to export control restrictions under the applicable laws of respective countries. Licensee shall fully comply with all applicable export license restrictions and requirements as well as with all lawses hereunder and shall procure all necessary governmental authorizations, including without limitation, all necessary licenses, approvals, permissions or consents, where necessary for the re-exportation of the Redistributables, Applications and/or Devices.

14.9 No Implied License
There are no implied licenses or other implied rights granted under this Agreement, and all rights, save for those expressly granted hereunder, shall remain with The Qt Company and its licensors. In addition, no licenses or immunities are granted to the combination of the Licensed Software with any other software or hardware not delivered by The Qt Company under this Agreement.

14.10 Attorney Fees
The prevailing Party in any action to enforce this Agreement shall be entitled to recover its attorney's fees and costs in connection with such action.

14.11 Severability

If any provision of this Agreement shall be adjudged by any court of competent jurisdiction to be unenforceable or invalid, that provision shall be limited or
eliminated to the minimum extent necessary so that this Agreement shall otherwise remain in full force and effect and enforceable.

IN WITNESS WHEREOF, the Parties hereto, intending to be legally bound hereby, have caused this Agreement to be executed by Licensees authorized representative installing the Licensed Software and accepting the terms hereof in connection therewith.

Appendix 1

1. Parts of the Licensed Software that are permitted for distribution in object code form only ("Redistributables") under this Agreement:

- The Licensed Software's essential and add-on libraries

- The Licensed Software's configuration tool ("qtconfig")

- The Licensed Software's help tool ("Qt Assistant")

- The Licensed Software's internationalization tools ("Qt Linguist", "lupdate", "lrelease")

- The Licensed Software's QML ("Qt Quick") launcher tool ("qmlscene” and "qmlviewer")

- The Licensed Software's installer framework

2. Parts of the Licensed Software that are not permitted for distribution include, but are not limited to:

- The Licensed Software's source code and header files

- The Licensed Software's documentation

- The Licensed Software's documentation generation tool ("qdoc")

- The Licensed Software's tool for writing makefiles ("qmake")

- The Licensed Software's Meta Object Compiler ("moc")

- The Licensed Software's User Interface Compiler ("uic” or in the case of Qt Jambi: "juic")

- The Licensed Software's Resource Compiler ("rcc")

- The Licensed Software's generator (only in the case of Qt Jambi if applicable)

- The Licensed Software's parts of the IDE tool ("Qt Creator")
- The Licensed Software's Emulator

- Build scripts, recipes and other material for creating the configuration of Licensed Software and/or 3rd party components, including the reference operating system configuration delivered in conjunction with the

The ISC License

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2015, Rebecca Turner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

NPM License Checker


*As of v17.0.0 the `failOn` and `onlyAllow` arguments take semicolons as delimeters instead of commas. Some license names contain commas and it messed with the parsing*

Ever needed to see all the license info for a module and its dependencies?

It's this easy:
```
```
```shell	npm install -g license-checker

mkdir foo
cd foo
npm install yui-lint
license-checker
```
You could see something like this:

```bash
---
debug@2.0.0
   repository: https://github.com/visionmedia/debug
   licenses: MIT*
---
```

Options
-------

* `--production` only show production dependencies.
* `--development` only show development dependencies.
* `--start [path of the initial json to look for]`
* `--unknown` report guessed licenses as unknown licenses.
* `--onlyunknown` only list packages with unknown or guessed licenses.
* `--json` output in json format.
* `--csv` output in csv format.
* `--csvComponentPrefix` prefix column for component in csv format.
* `--out [filepath]` write the data to a specific file.
* `--customPath` to add a custom Format file in JSON
* `--exclude [list]` exclude modules which licenses are in the comma-separated list from the output
* `--relativeLicensePath` output the location of the license files as relative paths
* `--summary` output a summary of the license usage`
* `--failOn [list]` fail (exit with code 1) on the first occurrence of the licenses of the semicolon-separated list
* `--onlyAllow [list]` fail (exit with code 1) on the first occurrence of the licenses not in the semicolon-separated list
* `--packages [list]` restrict output to the packages (package@version) in the semicolon-separated list
* `--excludePackages [list]` restrict output to the packages (package@version) not in the semicolon-separated list
* `--excludePrivatePackages` restrict output to not include any package marked as private
* `--direct` look for direct dependencies only`  

Exclusions
-----------

A list of licenses is the simplest way to describe what you want to exclude.

You can use valid [SPDX identifiers](https://spdx.org/licenses/).
You can use valid SPDX expressions like `MIT OR X11`.
You can use non-valid SPDX identifiers, like `Public Domain`, since `npm` does support some license strings that are not SPDX identifiers.

Examples
--------

```bash
---
license-checker --json > /path/to/licenses.json
license-checker --csv --out /path/to/licenses.csv
license-checker --unknown
license-checker --customPath customFormatExample.json
```
license-checker --exclude 'MIT, MIT OR X11, BSD, ISC'
license-checker --packages 'react@16.3.0;react-dom@16.3.0;lodash@4.3.1'
license-checker --excludePackages 'internal-1;internal-2'
license-checker --onlyunknown
```

Custom format
--------------

The `--customPath` option can be used with CSV to specify the columns. Note that the first column, `module_name`, will always be used.

When used with JSON format, it will add the specified items to the usual ones.

The available items are the following:
- name
- version
- description
- repository
- publisher
- email
- url
- licenses
- licenseFile
- licenseText
- licenseModified

You can also give default values for each item.
See an example in [customFormatExample.json](customFormatExample.json).

Requiring
---------
```
```
Debugging
--------

license-checker uses [debug](https://www.npmjs.com/package/debug) for internal logging. There are two internal markers:

* `license-checker:error` for errors
* `license-checker:log` for non-errors

Set the `DEBUG` environment variable to one of these to see debug output:

```
$ export DEBUG=license-checker*; license-checker
scanning ./yui-lint
cli@0.4.3
  repository: http://github.com/chriso/cli
  licenses: MIT
# ...
```

How Licenses are Found
-----------------------

We walk through the `node_modules` directory with the [`read-installed`](https://www.npmjs.org/package/read-installed) module. Once we gathered a list of modules we walk through them and look at all of their `package.json`’s. We try to identify the license with the [`spdx`](https://www.npmjs.com/package/spdx) module to see if it has a valid SPDX license attached. If that fails, we then look into the module for the following files: `LICENSE`, `LICENCE`, `COPYING`, & `README`.

If one of the those files are found (in that order) we will attempt to parse the license data from it with a list of known license texts. This will be shown with the `*` next to the name of the license to show that we “guessed” at it.

The Mesa header files use the following licenses.

================================================================================
The default Mesa license is as follows:

Copyright (C) 1999-2007 Brian Paul  All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

================================================================================

GLES/glext.h, GLES/gl.h and GLES/glplatform.h use the following license:

SGI FREE SOFTWARE LICENSE B
(Version 2.0, Sept. 18, 2008)

Copyright (C) [dates of first publication] Silicon Graphics, Inc. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice including the dates of first publication and either this permission notice or a reference to http://oss.sgi.com/projects/FreeB/ shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL SILICON GRAPHICS, INC. BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Silicon Graphics, Inc. shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from Silicon Graphics, Inc.

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This BSD License conforms to the Open Source Initiative "Simplified BSD License" as published at:
http://www.opensource.org/licenses/bsd-license.php

IAccessible2 is a trademark of the Linux Foundation. The IAccessible2 mark may be used in accordance with the Linux Foundation Trademark Policy to indicate compliance with the IAccessible2 specification.

************************************************************************/

<!-- LICENSEFILE/ -->

<h1>License</h1>

Unless stated otherwise all works are:

<ul><li>Copyright &copy; 2013+ <a href="http://bevry.me">Bevry Pty Ltd</a></li></ul>

and licensed under:

<ul><li><a href="http://spdx.org/licenses/MIT.html">MIT License</a></li></ul>

<h2>MIT License</h2>

<pre>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
</pre>
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2016 Peter Goldsborough

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Font license info

## Elusive

Copyright (C) 2013 by Aristeides Stathopoulos

Author: Aristeides Stathopoulos
License: SIL (http://scripts.sil.org/OFL)
Homepage: http://aristeides.com/

/*
 * $Id: pa_memorybarrier.h 1240 2007-07-17 13:05:07Z bjornroche $
 * Portable Audio I/O Library
 * Memory barrier utilities
 *
 * Author: Bjorn Roche, XO Audio, LLC
 *
 * This program uses the PortAudio Portable Audio Library.
 * For more information see: http://www.portaudio.com
 * Copyright (c) 1999-2000 Ross Bencina and Phil Burk
 */
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The text above constitutes the entire PortAudio license; however, the PortAudio community also makes the following non-binding requests:

Any person wishing to distribute modifications to the Software is requested to send the modifications to the original developer so that they can be incorporated into the canonical version. It is also requested that these non-binding requests be included along with the license above.

This program uses the PortAudio Portable Audio Library. For more information see: http://www.portaudio.com

Copyright (c) 1999-2000 Ross Bencina and Phil Burk
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The text above constitutes the entire PortAudio license; however, the PortAudio community also makes the following non-binding requests:

Any person wishing to distribute modifications to the Software is requested to send the modifications to the original developer so that they can be incorporated into the canonical version. It is also requested that these non-binding requests be included along with the license above.

The Catharon Open Source LICENSE
---------------------------------

2000-Jul-04

Copyright (C) 2000 by Catharon Productions, Inc.

Introduction
============

This license applies to source files distributed by Catharon Productions, Inc. in several archive packages. This license applies to all files found in such packages which do not fall under their own explicit license.
This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we are interested in any kind of bug reports. (‘as is’ distribution)

- You can use this software for whatever you want, in parts or full form, without having to pay us. (‘royalty-free’ usage)

- You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the Catharon Code. (‘credits’)

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering the packages distributed by Catharon Productions, Inc. and assume no liability related to their use.

Legal Terms
============

0. Definitions
--------------

Throughout this license, the terms ‘Catharon Package’, ‘package’, and ‘Catharon Code’ refer to the set of files originally distributed by Catharon Productions, Inc.

‘You’ refers to the licensee, or person using the project, where ‘using’ is a generic term including compiling the project’s source code as well as linking it to form a ‘program’ or ‘executable’. This program is referred to as ‘a program using one of the Catharon Packages’.

This license applies to all files distributed in the original Catharon Package(s), including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The Catharon Packages are copyright (C) 2000 by Catharon Productions, Inc. All rights reserved except as specified below.
1. No Warranty

-------------

THE CATHARON PACKAGES ARE PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OF OR THE INABILITY TO USE THE CATHARON PACKAGE.

2. Redistribution

-------------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the Catharon Packages (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file ("license.txt") unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that states that the software is based in part on the work of Catharon Productions, Inc. in the distribution documentation.

These conditions apply to any software derived from or based on the Catharon Packages, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

-------------

Neither Catharon Productions, Inc. and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use the following phrase to refer to this software in your documentation: 'this software is based in part on the Catharon Typography Project'.
As you have not signed this license, you are not required to accept it. However, as the Catharon Packages are copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the Catharon Packages, you indicate that you understand and accept all the terms of this license.

--- end of license.txt ---

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright contributors

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

PCRE is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

This is JavaScriptCore's variant of the PCRE library. While this library
started out as a copy of PCRE, many of the features of PCRE have been
removed.

Copyright (c) 1997-2005 University of Cambridge. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Apple
  Inc. nor the names of their contributors may be used to endorse or
  promote products derived from this software without specific prior
  written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2016, Vladimir Agafonkin

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The XML testsuite available here is a copy of the Extensible Markup Language (XML) Conformance Test Suites provided by W3C. Please see http://www.w3.org/XML/Test/ for updates and other information.

These files are licensed under the W3C Software License (19980720), reproduced below:

---

W3C SOFTWARE NOTICE AND LICENSE


This W3C work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body of any redistributed or derivative code:

3. Notice of any changes or modifications to the W3C files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

The MIT License

Copyright (c) 2015 Kyle E. Mitchell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2006 Hewlett-Packard Development Company, L.P.
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the
Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF
CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Fonts are (c) Bitstream (see below). DejaVu changes are in public domain.
Glyphs imported from Arev fonts are (c) Tavmjong Bah (see below)

Bitstream Vera Fonts Copyright
-------------------------------
Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is
a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of the fonts accompanying this license ("Fonts") and associated
documentation files (the "Font Software"), to reproduce and distribute the
Font Software, including without limitation the rights to use, copy, merge,
publish, distribute, and/or sell copies of the Font Software, and to permit
persons to whom the Font Software is furnished to do so, subject to the
following conditions:

The above copyright and trademark notices and this permission notice shall
be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular
the designs of glyphs or characters in the Fonts may be modified and
additional glyphs or characters may be added to the Fonts, only if the fonts
are renamed to names not containing either the words "Bitstream" or the word
"Vera".

This License becomes null and void to the extent applicable to Fonts or Font
Software that has been modified and is distributed under the "Bitstream Vera" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of Gnome, the Gnome Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the Gnome Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

Arev Fonts Copyright

Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the modifications to the Bitstream Vera Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Tavmjong Bah" or the word "Arev".

This License becomes null and void to the extent applicable to Fonts...
or Font Software that has been modified and is distributed under the "Tavmjong Bah Arev" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL TAVMJONG BAH BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the name of Tavmjong Bah shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from Tavmjong Bah. For further information, contact: tavmjong @ free . fr.

$Id: LICENSE 2133 2007-11-28 02:46:28Z lechimp $
Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2015 Nathan Rajlich

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

See Terms of Use for definitions of Unicode Inc.'s
Data Files and Software.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S
DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),
YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE
TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE
THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2019 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.
// Copyright 2019 The Chromium OS Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2012-2017 by Ingvar Stepanyan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Kael Zhang <i@kael.me>, contributors
http://kael.me/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright 2010 James Halliday (mail@substack.net)
Modified work Copyright 2014 Contributors (ben@npmjs.com)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2018, Gajus Kuizinas (http://gajus.com/)
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Gajus Kuizinas (http://gajus.com/) nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ANUARY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The OpenGL Extension Wrangler Library
Copyright (C) 2002-2007, Milan Ikits <milan ikits[.]ieee org>
Copyright (C) 2002-2007, Marcelo E. Magallon <mmagallo[.]debian org>
Copyright (C) 2002, Lev Povalahev
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The name of the author may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Mesa 3-D graphics library
Version: 7.0

Copyright (C) 1999-2007  Brian Paul  All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2007 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

MIT License

Copyright (c) 2013 Julian Gruber <julian@juliangruber.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 Gary Hsu

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2011-2018, Christopher Jeffrey (https://github.com/chjj/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2016 the fiat-crypto authors (see https://github.com/mit-plv/ fiat-crypto/blob/master/AUTHORS).
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2007, 2013 Apple Inc. All rights reserved.
Copyright (C) Rich Moore. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE COMPUTER, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE COMPUTER, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

string-unpack-code.js contains code from jQuery 1.2.1 - New Wave Javascript:

Copyright (c) 2007 John Resig (jquery.com)
Dual licensed under the MIT (MIT-LICENSE.txt)
and GPL (GPL-LICENSE.txt) licenses.

and

Copyright (c) 2004-2007, The Dojo Foundation
All Rights Reserved.

Licensed under the Academic Free License version 2.1 or above OR the
modified BSD license. For more information on Dojo licensing, see:

http://dojotoolkit.org/community/licensing.shtml

and

Prototype 1.5 rc0
- Adapted from Ruby on Rails - http://dev.rubyonrails.org/browser/spinoffs/prototype/src
- By Lunarmedia, 06 August, 2006
- Available at (and packed with) JavascriptCompressor.com

string-tagcloud.js contains code from json.js 2007-10-10, placed in the Public Domain

string-base64.js contains code licensed under MPL 1.1 OR GPL 2.0 OR LGPL 2.1
The Initial Developer of the Original Code is Digital Creations 2, Inc.
Portions created by the Initial Developer are Copyright (C) 2000
the Initial Developer. All Rights Reserved.

Contributor(s):
* Martijn Pieters <mj@digicool.com> (original author)
* Samuel Sieb <samuel@sieb.net>

date-format-xparb.js is licensed under the LGPL v2.1.
Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and
ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:
a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

Copyright (c) 2011-2015 Jake Luer jake@alogicalparadox.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2009 by the Jinja Team, see AUTHORS for more details.

Some rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2013-2018 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (C) 2002-2013 The ANGLE Project Authors.
// All rights reserved.
//
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of TransGaming Inc., Google Inc., 3DLabs Inc. Ltd., nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This Font Software is licensed under the SIL Open Font License, Version 1.1.

This license is copied below, and is also available with a FAQ at: http://scripts.sil.org/OFL

-----------------------------------------------------------
SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007
-----------------------------------------------------------

PREAMBLE

The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The
fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font
Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
`cssmin.py` - A Python port of the YUI CSS compressor.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This software contains portions of the YUI CSS Compressor, notably some regular expressions for reducing the size of CSS. The YUI Compressor source code can be found at <http://github.com/yui/yuicompressor>, and is licensed as follows:

> YUI Compressor Copyright License Agreement (BSD License)
> Copyright (c) 2009, Yahoo! Inc.
> All rights reserved.
> Redistribution and use of this software in source and binary forms,
> with or without modification, are permitted provided that the following
> conditions are met:
> * Redistributions of source code must retain the above
copyright notice, this list of conditions and the
following disclaimer.
> * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other
materials provided with the distribution.
> * Neither the name of Yahoo! Inc. nor the names of its
> contribute may be used to endorse or promote products
> derived from this software without specific prior
> written permission of Yahoo! Inc.
> THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
> AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
> IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
> DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
> FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
> DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
> SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
> CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
> OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
> OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the

Open Source Used In DNAC Platform Arica 5855
notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

   The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

   If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

MIT License

Copyright (c) 2017 itmeo

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
// Copyright (c) 2011 Google, Inc.
//
// Permission is hereby granted, free of charge, to any person obtaining a copy
// of this software and associated documentation files (the "Software"), to deal
// in the Software without restriction, including without limitation the rights
// to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
// copies of the Software, and to permit persons to whom the Software is
// furnished to do so, subject to the following conditions:
//

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter, Ben Noordhuis, and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

Copyright (c) 2011 The FreeBSD Foundation
All rights reserved.
Portions of this software were developed by David Chisnall under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,

notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
The ISC License

Copyright (c) 2019 Elan Shanker, Paul Miller (https://paulmillr.com)

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Node.js is licensed for use as follows:

---

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

This license applies to parts of Node.js originating from the
https://github.com/joyent/node repository:

Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

Notice that the following BSD-style license applies to this one
file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

----------------------------------------------------------------

This file is part of Valgrind, a dynamic binary instrumentation framework.

Copyright (C) 2000-2010 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----------------------------------------------------------------

Notice that the above BSD-style license applies to this one file (valgrind.h) only. The entire rest of Valgrind is licensed under the terms of the GNU General Public License, version 2. See the
COPYING file in the source distribution for details.

-------------------------------------------------------------------

The MIT License (MIT)

Copyright (c) 2018 Tobias Reich

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

This package was debianized by gperftools Contributors <google-perftools@googlegroups.com>

It was downloaded from http://code.google.com/p/gperftools/downloads/list

Upstream Author: google-perftools@googlegroups.com

Copyright (c) 2005, Google Inc.
All rights reserved.

Copyright (c) 2013, gperftools Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2014, Michael Bostock
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name Michael Bostock may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL MICHAEL BOSTOCK BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those
products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". Licensees and recipients may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible
feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.
2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work’s users, your or third parties’ legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any
You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.
A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.
"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains
a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However,
nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to
make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting
any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different
15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
Copyright (c) 2013, Joel Feenstra
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the ESQuery nor the names of its contributors may
be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL JOEL FEENSTRA BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------

Copyright (c) 2011-2015 Jake Luer jake@alogicalparadox.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----

This library bundles a version of the `fs.realpath` and `fs.realpathSync` methods from Node.js v0.10 under the terms of the Node.js MIT license.

Node's license follows, also included at the header of `old.js` which contains the licensed code:

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
t editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

#!/usr/bin/env node

/*
Copyright (c) 2013, Yahoo! Inc. All rights reserved.
Code licensed under the BSD License:
http://yuilibrary.com/license/
*/

var checker = require('../lib/index');

var args = require('../lib/args').parse();
var mkdirp = require('mkdirp');
var path = require('path');
var chalk = require('chalk');
var fs = require('fs');

if (args.help) {
    console.error('license-checker@' + require('../package.json').version);
    var usage = [
        '    
        ',
        '    --production only show production dependencies.',
        '    --development only show development dependencies.',
        '    --unknown report guessed licenses as unknown licenses.',
        '    --start [path of the initial json to look for]',
        '    --onlyunknown only list packages with unknown or guessed licenses',
        '    --json output in json format.',
        '    --csv output in csv format.',
        '    --csvComponentPrefix column prefix for components in csv file',
        '    --out [filepath] write the data to a specific file.',
        '    --exclude [list] exclude modules which licenses are in the comma-separated list from the output',
    ];
    console.error(usage.join(' '));
}
'--relativeLicensePath output the location of the license files as relative paths',
'--summary output a summary of the license usage',
'--failOn [list] fail (exit with code 1) on the first occurrence of the licenses of the semicolon-separated list',
'--onlyAllow [list] fail (exit with code 1) on the first occurrence of the licenses not in the semicolon-separated list',
'--direct look for direct dependencies only',
'--packages [list] restrict output to the packages (package@version) in the semicolon-seperated list',
'--excludePackages [list] restrict output to the packages (package@version) not in the semicolon-seperated list',
'--excludePrivatePackages restrict output to not include any package marked as private',
'
   --version The current version',
'   --help The text you are reading right now :)',
'   ];
console.error(usage.join('
'), '
');
process.exit(0);
}

if (args.version) {
    console.error(require('../package.json').version);
    process.exit(1);
}

if (args.failOn && args.onlyAllow) {
    console.error('--failOn and --onlyAllow can not be used at the same time. Choose one or the other.');
    process.exit(1);
} else {
    var argValue = args.failOn || args.onlyAllow;
    if (argValue && argValue.indexOf(',') >= 0) {
        var argName = args.failOn ? 'failOn' : 'onlyAllow';
        console.warn('Warning: As of v17 the --' + argName + ' argument takes semicolons as delimeters instead of commas (some license names can contain commas)');
    }
}

checker.init(args, function(err, json) {

    var formattedOutput = "\n";

    if (!err) {
        console.error('Found error');
        console.error(err);
    }

    if (shouldColorizeOutput(args)) {
        var keys = Object.keys(json);
        keys.forEach(function(key) {
var keyParts = key.split('@');
var colorizedKey = chalk.blue(keyParts[0]) + chalk.dim('@') + chalk.green(keyParts[1]);
json[colorizedKey] = json[key];
delete json[key];
});
}

if (args.json) {
  formattedOutput = JSON.stringify(json, null, 2) + '\n';
} else if (args.csv) {
  formattedOutput = checker.asCSV(json, args.customFormat, args.csvComponentPrefix);
} else if (args.markdown) {
  formattedOutput = checker.asMarkDown(json, args.customFormat) + '\n';
} else if (args.summary) {
  formattedOutput = checker.asSummary(json);
} else {
  formattedOutput = checker.asTree(json);
}

if (args.files) {
  checker.asFiles(json, args.files);
} else if (args.out) {
  var dir = path.dirname(args.out);
  mkdirp.sync(dir);
  fs.writeFileSync(args.out, formattedOutput, 'utf8');
} else {
  console.log(formattedOutput);
}

function shouldColorizeOutput(args) {
  return args.color && !args.out && !(args.csv || args.json || args.markdown);
}

The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.
Uranus Ring:  
Used under the terms of the license at http://www.jpl.nasa.gov/imagepolicy/  
MIT License  

Copyright (c) 2017 crypto-browserify  

Permission is hereby granted, free of charge, to any person obtaining a copy  
of this software and associated documentation files (the "Software"), to deal  
in the Software without restriction, including without limitation the rights  
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell  
copies of the Software, and to permit persons to whom the Software is  
furnished to do so, subject to the following conditions:  

The above copyright notice and this permission notice shall be included in all  
copies or substantial portions of the Software.  

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR  
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,  
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE  
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER  
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,  
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE  
SOFTWARE.  
(The MIT License)  

Copyright (c) 2013 Roman Shtylman <shtylman@gmail.com>  

Permission is hereby granted, free of charge, to any person obtaining  
a copy of this software and associated documentation files (the  
'Software'), to deal in the Software without restriction, including  
without limitation the rights to use, copy, modify, merge, publish,  
distribute, sublicense, and/or sell copies of the Software, and to  
permit persons to whom the Software is furnished to do so, subject to  
the following conditions:  

The above copyright notice and this permission notice shall be  
included in all copies or substantial portions of the Software.  

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,  
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF  
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY  
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,  
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE  
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.  
The MIT License (MIT)
Copyright (c) 2015 Daniel Frey

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2007-2009 Google Inc.
Copyright 2007-2009 WebDriver committers

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The text and information contained in this file may be freely used, copied, or distributed without compensation or licensing restrictions.
// Copyright (c) 2014 The Polymer Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2017, Google Inc.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:
(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

// Copyright 2016 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

LICENSE

Lipi core toolkit is licensed under the MIT license.
- MIT-Style License
http://www.opensource.org/licenses/mit-license.php

This is the weakest license, meaning it applies the least restrictions. It is used for most parts of the Lipi Toolkit, and especially for libraries which are potentially included (by copying) into other libraries (as sub-components). The intention is to make sure the covered software can be used really everywhere and for every reason without any real restrictions.

Copyright (c) 2007-2014, Alexandru Marasteanu <hello [at] alexei [dot] ro>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of this software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MPL-1.1 / GPL-2.0 / LGPL-2.1

SkGifImageReader.cpp and SkGifImageReader.h:

***** BEGIN LICENSE BLOCK *****
Version: MPL 1.1/GPL 2.0/LGPL 2.1

The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the
The Original Code is mozilla.org code.

The Initial Developer of the Original Code is Netscape Communications Corporation.
Portions created by the Initial Developer are Copyright (C) 1998 the Initial Developer. All Rights Reserved.

Contributor(s):
Chris Saari <saari@netscape.com>
Apple Computer

Alternatively, the contents of this file may be used under the terms of either the GNU General Public License Version 2 or later (the "GPL"), or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), in which case the provisions of the GPL or the LGPL are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of either the GPL or the LGPL, and not to allow others to use your version of this file under the terms of the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL or the LGPL. If you do not delete the provisions above, a recipient may use your version of this file under the terms of any one of the MPL, the GPL or the LGPL.

***** END LICENSE BLOCK ***** */

BSD-3-Clause
============

libgifcodec.gni, SkGifCodec.h, SkLibGifCodec.cpp, SkLibGifCodec.h:

Copyright 2019 Google LLC. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BSD-2-Clause
============

SkLibGifCodec.cpp:

Copyright (C) 2006 Apple Computer, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE COMPUTER, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE COMPUTER, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The BSD License)

Copyright (c) 2010-2014, Christian Johansen, christian@cjohansen.no
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Christian Johansen nor the names of his contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright 2013 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
// *
//   * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
//   * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
//   * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY. WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2004, 2005, 2006 Ero Carrera <ero@dkbza.org>. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT
OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.

/* ***** BEGIN LICENSE BLOCK *****
* Version: MPL 1.1/GPL 2.0/LGPL 2.1
*
* The contents of this file are subject to the Mozilla Public License Version
* 1.1 (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* http://www.mozilla.org/MPL/
*
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*
* The Original Code is mozilla.org code.
*
* The Initial Developer of the Original Code is
* Netscape Communications Corporation.
* Portions created by the Initial Developer are Copyright (C) 2002
* the Initial Developer. All Rights Reserved.
*/
* Contributor(s):

* Alternatively, the contents of this file may be used under the terms of
* either the GNU General Public License Version 2 or later (the "GPL"), or
* the GNU Lesser General Public License Version 2.1 or later (the "LGPL"),
* in which case the provisions of the GPL or the LGPL are applicable instead
* of those above. If you wish to allow use of your version of this file only
* under the terms of either the GPL or the LGPL, and not to allow others to
* use your version of this file under the terms of the MPL, indicate your
* decision by deleting the provisions above and replace them with the notice
* and other provisions required by the GPL or the LGPL. If you do not delete
* the provisions above, a recipient may use your version of this file under
* the terms of any one of the MPL, the GPL or the LGPL.

* ****** END LICENSE BLOCK ****** */

QT LICENSE AGREEMENT
Agreement version 4.2.3

This Qt License Agreement ("Agreement") is a legal agreement for the licensing
of Licensed Software (as defined below) between The Qt Company (as defined
below) and the Licensee who has accepted the terms of this Agreement by
downloading or using the Licensed Software and/or as defined herein:

Capitalized terms used herein are defined in Section 1.

WHEREAS:
(A) Licensee wishes to use the Licensed Software for the purpose of
developing and distributing Applications and/or Devices (each as
defined below); and
(B) The Qt Company is willing to grant the Licensee a right to use
Licensed Software for such a purpose pursuant to term and conditions
of this Agreement.

NOW, THEREFORE, THE PARTIES HEREBY AGREE AS FOLLOWS:

1. DEFINITIONS

"Affiliate" of a Party shall mean an entity (i) which is directly
or indirectly controlling such Party; (ii) which is under the same direct or
indirect ownership or control as such Party; or (iii) which is directly or
indirectly owned or controlled by such Party. For these purposes, an entity
shall be treated as being controlled by another if that other entity has fifty
percent (50 %) or more of the votes in such entity, is able to direct its
affairs and/or to control the composition of its board of directors or
equivalent body.

"Add-on Products" shall mean The Qt Company's specific add-on software products
(for example Qt Safe Renderer, Qt for Automation, Qt Application Manager), which
are not licensed as part of The Qt Company's standard offering, but shall be included into the scope of Licensed Software only if so specifically agreed between the Parties.

"Applications" shall mean Licensee's software products created using the Licensed Software, which may include the Redistributables, or part thereof.

"Contractor(s)" shall mean third party consultants, distributors and contractors performing services to the Licensee under applicable contractual arrangement.

"Customer(s)" shall mean Licensee's end users to whom Licensee, directly or indirectly, distributes copies of the Redistributables.

"Data Protection Legislation" shall mean the General Data Protection Regulation (EU 2016/679) (GDPR) and any national implementing laws, regulations and secondary legislation, as may be amended or updated from time to time, as well as any other data protection laws or regulations applicable in relevant territory.

"Deployment Platforms" shall mean operating systems and/or hardware specified in the License Certificate, on which the Redistributables can be distributed pursuant to the terms and conditions of this Agreement.

"Designated User(s)" shall mean the employee(s) of Licensee or Licensee's Affiliates acting within the scope of their employment or Licensee's Contractors acting within the scope of their services for Licensee and on behalf of Licensee. Designated Users shall be named in the License Certificate.

"Development License" shall mean the license needed by the Licensee for each Designated User to use the Licensed Software under the license grant described in Section 3.1 of this Agreement. Development Licenses are available per respective Licensed Software products, each product having its designated scope and purpose of use. Distribution Licenses are always connected to Qt for Device Creation product and Qt for MCUs product only.

"Development Platforms" shall mean those operating systems specified in the License Certificate, in which the Licensed Software can be used under the Development License, but not distributed in any form or used for any other purpose.

"Devices" shall mean hardware devices or products that 1) are manufactured and/or distributed by the Licensee or its Affiliates or Contractors, and (2)(i) incorporate or integrate the Redistributables or parts thereof; or (ii) where the main user interface or substantial functionality of such unit, when used by a Customer, is provided by Application(s) or otherwise depends on the Licensed Software, regardless of whether the Redistributables are distributed together with the hardware or not. Devices covered with this Agreement shall be specified in Appendix 2 or in a quote.
"Distribution License(s)" shall mean the license required for any kind of sale, trade, exchange, loan, lease, rental or other distribution by or on behalf of Licensee to a third party of Redistributables in connection with Devices pursuant to license grant described in Section 3.3 of this Agreement.

"Distribution License Packs" shall mean set of prepaid Distribution Licenses for distribution of Redistributables, as defined in The Qt Company's standard price list, quote, Purchase Order confirmation or in an appendix hereto, as the case may be.

"Intellectual Property Rights" shall mean patents (including utility models), design patents, and designs (whether or not capable of registration), chip topography rights and other like protection, copyrights, trademarks, service marks, trade names, logos or other words or symbols and any other form of statutory protection of any kind and applications for any of the foregoing as well as any trade secrets.

"License Certificate" shall mean a certificate generated by The Qt Company for each Designated User respectively upon them downloading the Licensed Software, which will be available under respective Designated User's Qt Account at account.qt.io. License Certificates will specify the Designated User, the Development Platforms, Deployment Platforms and the License Term. Such terms are considered part of the licenses granted hereunder and shall be updated from time to time to reflect any agreed changes to the foregoing terms relating to Designated User's rights to the Licensed Software.

"License Fee" shall mean the fee charged to the Licensee for rights granted under the terms of this Agreement.

"License Term" shall mean the agreed validity period of the Development License of the respective Designated User, during which time the Designated User is entitled to use the Licensed Software, as set forth in the respective License Certificate.

"Licensed Software" shall mean either
(i) Qt for Application Development or
(ii) Qt for Device Creation, and/or
(iii) Qt 3D Studio, and/or
(iv) Qt Design Studio, and/or
(v) Qt for MCUs, and/or
(vi) selected Add-on Products, if any,

depending on which product(s) the Licensee has purchased under this Agreement, as well as corresponding online or electronic documentation, associated media and printed materials, including the source code (where applicable), example programs and the documentation, licensed to the Licensee under this Agreement. Licensed Software does not include Third Party Software (as defined in Section 4) or Open Source Qt. The Qt Company may, in the course of its
development activities, at its free and absolute discretion and without any obligation to send or publish any notifications to the Licensee or in general, make changes, additions or deletions in the components and functionalities of the Licensed Software, provided that no such changes, additions or deletions will affect the already released version of the Licensed Software, but only upcoming version(s).

"Licensee" shall mean the individual or legal entity that is party to this Agreement, as identified on the signature page hereof.

"Licensee's Records" shall mean books and records that are likely to contain information bearing on Licensee's compliance with this Agreement, Licensee's use of Open Source Qt and/or the payments due to The Qt Company under this Agreement, including, but not limited to user information, assembly logs, sales records and distribution records.

"Modified Software" shall have the meaning as set forth in Section 2.3.

"Online Services" shall mean any services or access to systems made available by The Qt Company to the Licensee over the Internet relating to the Licensed Software or for the purpose of use by the Licensee of the Licensed Software or Support. Use of any such Online Services is discretionary for the Licensee and some of them may be subject to additional fees.

"Open Source Qt" shall mean the non-commercial Qt computer software products, licensed under the terms of the GNU Lesser General Public License, version 2.1 or later ("LGPL") or the GNU General Public License, version 2.0 or later ("GPL"). For clarity, Open Source Qt shall not be provided nor governed under this Agreement.

"Party" or "Parties" shall mean Licensee and/or The Qt Company.

"Permitted Software" shall mean any (i) open source software (excluding Open Source Qt) that is a) generally available for public in source code form without additional cost under any of the licenses approved by Open Source Initiative as listed on https://opensource.org/licenses; and (b) is in no way, directly or indirectly, developed by or for or otherwise related to or in the interest of the Licensee or its Affiliates, or (ii) software The Qt Company has made available via its Qt Marketplace online distribution channel.

"Pre-Release Code" shall have the meaning as set forth in Section 4.

"Prohibited Combination" shall mean any means to (i) use, combine, incorporate, link or integrate Licensed Software with any software created with or incorporating Open Source Qt, (ii) use Licensed Software for creation of any software created with or incorporating Open Source Qt, or (iii) incorporate or integrate Applications into a hardware device or product other than a Device.
"Qt 3D Studio" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt Design Studio" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for Application Development" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for Device Creation" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for MCUs" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Redistributables" shall mean the portions of the Licensed Software set forth in Appendix 1 that may be distributed pursuant to the terms of this Agreement in object code form only, including any relevant documentation. Where relevant, any reference to Licensed Software in this Agreement shall include and refer also to Redistributables.

"Renewal Term" shall mean an extension of previous License Term as agreed between the Parties.

"Submitted Modified Software" shall have the meaning as set forth in Section 2.3.

"Support" shall mean standard developer support that is provided by The Qt Company to assist Designated Users in using the Licensed Software in accordance with The Qt Company's standard support terms available at https://www.qt.io/terms-conditions/ and as further defined in Section 8 hereunder.

"Taxes" shall have the meaning set forth in Section 10.5.

"Term" shall have the meaning set forth in Section 12.

"The Qt Company" shall mean:

(i) in the event Licensee is an individual residing in the United States or a legal entity incorporated in the United States or having its headquarters in the United States, The Qt Company Inc., a Delaware corporation with its office at 2350 Mission College Blvd., Suite 1020, Santa Clara, CA 95054, USA.; or

(ii) in the event the Licensee is an individual residing outside of the United States or a legal entity incorporated outside of the United States or having its registered office outside of the United States,
"Third-Party Software" shall have the meaning set forth in Section 4.

"Updates" shall mean a release or version of the Licensed Software containing bug fixes, error corrections and other changes that are generally made available to users of the Licensed Software that have contracted for Support. Updates are generally depicted as a change to the digits following the decimal in the Licensed Software version number. The Qt Company shall make Updates available to the Licensee under the Support. Updates shall be considered as part of the Licensed Software hereunder.

"Upgrades" shall mean a release or version of the Licensed Software containing enhancements and new features and are generally depicted as a change to the first digit of the Licensed Software version number. In the event Upgrades are provided to the Licensee under this Agreement, they shall be considered as part of the Licensed Software hereunder.

2. OWNERSHIP

2.1. Ownership of The Qt Company
The Licensed Software is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Licensed Software is licensed, not sold.

All of The Qt Company's Intellectual Property Rights are and shall remain the exclusive property of The Qt Company or its licensors respectively.

2.2. Ownership of Licensee
All the Licensee's Intellectual Property Rights are and shall remain the exclusive property of the Licensee or its licensors respectively.

All Intellectual Property Rights to the Modified Software, Applications and Devices shall remain with the Licensee and no rights thereto shall be granted by the Licensee to The Qt Company under this Agreement (except as set forth in Section 2.3 below).

2.3. Modified Software
Licensee may create bug-fixes, error corrections, patches or modifications to the Licensed Software ("Modified Software"). Such Modified Software may break the source or binary compatibility with the Licensed Software (including without limitation through changing the application programming interfaces ("API") or by adding, changing or deleting any variable, method, or class signature in the Licensed Software and/or any inter-process protocols, services or standards in the Licensed Software libraries). To the extent that Licensee's Modified Software so breaks source or binary compatibility with the Licensed Software, Licensee acknowledges that The Qt Company's ability to provide Support may be
Licensee may, at its sole and absolute discretion, choose to submit Modified Software to The Qt Company ("Submitted Modified Software") in connection with Licensee's Support request, service request or otherwise. In the event Licensee does so, then, Licensee hereby grants The Qt Company a sublicensable, assignable, irrevocable, perpetual, worldwide, non-exclusive, royalty-free and fully paid-up license, under all of Licensee's Intellectual Property Rights, to reproduce, adapt, translate, modify, and prepare derivative works of, publicly display, publicly perform, sublicense, make available and distribute such Submitted Modified Software as The Qt Company sees fit at its free and absolute discretion.

3. LICENSES GRANTED

3.1. Development with Licensed Software
Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable license, valid for the License Term, to use, modify and copy the Licensed Software by Designated Users on the Development Platforms for the sole purposes of designing, developing, demonstrating and testing Application(s) and/or Devices, and to provide thereto related support and other related services to end-user Customers. Each Application and/or Device can only include, incorporate or integrate code generated under this Agreement by such Designated Users who are duly licensed for the applicable Development Platform(s) and Deployment Platform(s) (i.e. have a valid license for the appropriate Licensed Software product).

Licensee may install copies of the Licensed Software on five (5) computers per Designated User, provided that only the Designated Users who have a valid Development License may use the Licensed Software.

Licensee may at any time designate another Designated User to replace a then-current Designated User by notifying The Qt Company in writing, provided that any Designated User may be replaced only once during any six-month period.

Upon expiry of the initially agreed License Term, the respective License Terms shall be automatically extended to one or more Renewal Term(s), unless and until either Party notifies the other Party in writing that it does not wish to continue the License Term, such notification to be provided to the other Party no less than thirty (30) days before expiry of the respective License Term.

Unless otherwise agreed between the Parties, Renewal Term shall be of equal length with the initial License Term.

Any such Renewal Term shall be subject to License Fees agreed between the Parties or, if no advance agreement exists, subject to The Qt Company's standard pricing applicable at the commencement date of any such Renewal Term.
Any price or other term specified for a Renewal Term shall be valid only for the specified time.

The Qt Company may request the Licensee to place a purchase order corresponding to a quote by The Qt Company for the relevant Renewal Term.

In the event Licensee does not prevent auto-renewal pursuant the above, but a Renewal Term is nevertheless not duly ordered within 30 days from the date of the respective notification from The Qt Company and/or the respective License Fee paid by due date specified in The Qt Company’s respective invoice, The Qt Company shall apply a reinstatement fee equal to ten percent (10%) of the total value of the License Fees of the Development Licenses for the expired term to be added to the License Fee of the respective Renewal Term.

In the event Licensee chooses not to renew a Development License for a Renewal Term by notifying The Qt Company thereof no less than thirty (30) days before expiry of the respective License Term, Licensee may still reinstate such expired Development Licenses for a Renewal Term subject to applicable renewal Term License Fees until thirty (30) days from the expiry of the initially agreed License Term or preceding Renewal Term. After such thirty (30) day period a Development License shall be subject to applicable License Fees for a new Development License and not any Renewal Term License Fees.

3.2. Distribution of Applications
Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable, revocable (for cause pursuant to this Agreement) right and license, valid for the Term, to
(i) distribute, by itself or through its Contractors, Redistributables as installed, incorporated or integrated into Applications for execution on the Deployment Platforms, and
(ii) grant sublicenses to Redistributables, as distributed hereunder, for Customers solely for Customer’s internal use and to the extent necessary in order for the Customers to use the Applications for their respective intended purposes.

Right to distribute the Redistributables as part of an Application as provided herein is not royalty-bearing but is conditional upon the Licensee not having any unpaid License Fees for Development Licenses owed to The Qt Company at the time of distribution of any Redistributables to Customers.

3.3. Distribution of Devices
Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable, revocable (for cause pursuant to this Agreement) right and license, valid for the Term, to
(i) distribute, by itself or through one or more tiers of Contractors, Redistributables as installed, incorporated or integrated, or intended to be installed, incorporated or integrated into Devices for execution
on the Deployment Platforms, and
(ii) grant sublicenses to Redistributables, as distributed hereunder, for
Customers solely for Customer's internal use and to the extent
necessary in order for the Customers to use the Devices for their
respective intended purposes.

Right to distribute the Redistributables with Devices as provided herein is
conditional upon the Licensee (i) not having any unpaid License Fees for
Development Licenses owed to The Qt Company, and (ii) having purchased and paid
the corresponding Distribution Licenses at the time of distribution of any
Redistributables to Customers.

3.4. Further Requirements
The licenses granted above in this Section 3 by The Qt Company to Licensee are
conditional and subject to Licensee's compliance with the following terms:
(i) Licensee shall not remove or alter any copyright, trademark or other
proprietary rights notice(s) contained in any portion of the Licensed
Software;
(ii) Applications must add primary and substantial functionality to the
Licensed Software so as not to compete with the Licensed Software;
(iii) Applications may not pass on functionality which in any way makes it
possible for others to create software with the Licensed Software;
        provided however that Licensee may use the Licensed Software's
        scripting and QML ("Qt Quick") functionality solely in order to
        enable scripting, themes and styles that augment the functionality
        and appearance of the Application(s) without adding primary and
        substantial functionality to the Application(s);
(iv) Licensee shall not use Licensed Software in any manner or for any
        purpose that infringes, misappropriates or otherwise violates any
        Intellectual property or right of any third party, or that violates
        any applicable law;
(v) Licensee shall not use The Qt Company's or any of its suppliers' names,
        logos, or trademarks to market Applications, except that
        Licensee may use "Built with Qt" logo to indicate that
        Application(s) was developed using the Licensed Software;
(vi) Licensee shall not distribute, sublicense or disclose source code
        of Licensed Software to any third party (provided however that
        Licensee may appoint employee(s) of Contractors as Designated
        Users to use Licensed Software pursuant to this Agreement). Such
        right may be available for the Licensee subject to a separate
        software development kit ("SDK") license agreement to be concluded
        with The Qt Company;
(vii) Licensee shall not grant the Customers a right to (i) make copies of
        the Redistributables except when and to the extent required to use the
        Applications and/or Devices for their intended purpose, (ii) modify
        the Redistributables or create derivative works thereof, (iii)
        decompile, disassemble or otherwise reverse engineer Redistributables,
        or (iv) redistribute any copy or portion of the Redistributables to
(viii) Licensee shall not and shall cause that its Affiliates or Contractors shall not use Licensed Software in any Prohibited Combination, unless Licensee has received an advance written permission from The Qt Company to do so. Absent such written permission, any and all distribution by the Licensee during the Term of a hardware device or product a) which incorporate or integrate any part of Licensed Software or Open Source Qt; or b) where the main user interface or substantial functionality is provided by software built with Licensed Software or Open Source Qt or otherwise depends on the Licensed Software or Open Source Qt, shall be considered to be Device distribution under this Agreement and shall be dependent on Licensee's compliance thereof (including but not limited to obligation to pay applicable License Fees for such distribution). Notwithstanding what is provided above in this sub-section (viii), Licensee is entitled to use and combine Licensed Software with any Permitted Software;

(ix) Licensee shall cause all of its Affiliates and Contractors entitled to make use of the licenses granted under this Agreement, to be contractually bound to comply with the relevant terms of this Agreement and not to use the Licensed Software beyond the terms hereof and for any purposes other than operating within the scope of their services for Licensee. Licensee shall be responsible for any and all actions and omissions of its Affiliates and Contractors relating to the Licensed Software and use thereof (including but not limited to payment of all applicable License Fees);

(x) Except when and to the extent explicitly provided in this Section 3, Licensee shall not transfer, publish, disclose, display or otherwise make available the Licensed Software; and

(xi) Licensee shall not attempt or enlist a third party to conduct or attempt to conduct any of the above.

Above terms shall not be applicable if and to the extent they conflict with any mandatory provisions of any applicable laws.

Any use of Licensed Software beyond the provisions of this Agreement is strictly prohibited and requires an additional license from The Qt Company.

4. THIRD-PARTY SOFTWARE

The Licensed Software may provide links to third party libraries or code (collectively "Third-Party Software") to implement various functions. Third-Party Software does not comprise part of the Licensed Software. In some cases, access to Third-Party Software may be included with the Licensed Software. Such Third-Party Software will be listed in the "../src/3rdparty" source tree delivered with the Licensed Software or documented in the Licensed Software, as such may be amended from time to time. Licensee acknowledges that use or distribution of Third-Party Software is in all respects subject to
5. PRE-RELEASE CODE

The Licensed Software may contain pre-release code and functionality, or sample code marked or otherwise stated with appropriate designation such as "Technology Preview", "Alpha", "Beta", "Sample" etc. ("Pre-Release Code").

Such Pre-Release Code may be present complimentary for the Licensee, in order to provide experimental support or information for new platforms or preliminary versions of one or more new functionalities or for other similar reasons. The Pre-Release Code may not be at the level of performance and compatibility of a final, generally available, product offering. The Pre-Release Code may not operate correctly, may contain errors and may be substantially modified by The Qt Company prior to the first commercial product release, if any. The Qt Company is under no obligation to make Pre-Release Code commercially available, or provide any Support or Updates relating thereto. The Qt Company assumes no liability whatsoever regarding any Pre-Release Code, but any use thereof is exclusively at Licensee's own risk and expense.

For clarity, unless Licensed Software specifies different license terms for the respective Pre-Release Code, the Licensee is entitled to use such pre-release code pursuant to Section 3, just like other Licensed Software, provided however that in the event Add-on Products are included and available as such Pre-Release Code, Licensee's right to use such Add-on Products is nevertheless subject to and conditional upon conclusion of separate agreement with The Qt Company.

6. LIMITED WARRANTY AND WARRANTY DISCLAIMER

The Qt Company hereby represents and warrants that it has the power and authority to grant the rights and licenses granted to Licensee under this Agreement.

Except as set forth above, the Licensed Software is licensed to Licensee "as is" and Licensee's exclusive remedy and The Qt Company's entire liability for errors in the Licensed Software shall be limited, at The Qt Company's option, to correction of the error, replacement of the Licensed Software or return of the applicable fees paid for the defective Licensed Software for the time period during which the License is not able to utilize the Licensed Software under the terms of this Agreement.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE QT COMPANY ON BEHALF OF ITSELF AND ITS LICENSORS, SUPPLIERS AND AFFILIATES, DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT WITH REGARD TO THE LICENSED SOFTWARE. THE QT COMPANY DOES NOT WARRANT THAT THE LICENSED SOFTWARE WILL SATISFY LICENSEEE'S REQUIREMENTS OR
THAT IT WILL OPERATE WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED.

7. INDEMNIFICATION AND LIMITATION OF LIABILITY

7.1. Limitation of Liability
EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, AND (II) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY BE LIABLE TO THE OTHER PARTY FOR ANY LOSS OF PROFIT, LOSS OF DATA, LOSS OF BUSINESS OR GOODWILL OR ANY OTHER INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE COST, DAMAGES OR EXPENSE OF ANY KIND, HOWEVER ARISING UNDER OR IN CONNECTION WITH THIS AGREEMENT.

EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, AND (II) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY’S TOTAL AGGREGATE LIABILITY UNDER THIS AGREEMENT EXCEED THE AGGREGATE LICENSE FEES PAID OR PAYABLE TO THE QT COMPANY FROM LICENSEE DURING THE PERIOD OF TWELVE (12) MONTHS IMMEDIATELY PRECEDING THE EVENT RESULTING IN SUCH LIABILITY.

THE PROVISIONS OF THIS SECTION 7 ALLOCATE THE RISKS UNDER THIS AGREEMENT BETWEEN THE QT COMPANY AND LICENSEE AND THE PARTIES HAVE RELIED UPON THE LIMITATIONS SET FORTH HEREIN IN DETERMINING WHETHER TO ENTER INTO THIS AGREEMENT.

NOTWITHSTANDING ANYTHING TO THE CONTRARY IN THIS AGREEMENT, LICENSEE SHALL ALWAYS BE LIABLE TO PAY THE APPLICABLE LICENSE FEES CORRESPONDING TO ITS ACTUAL USE OF LICENSED SOFTWARE.

8. SUPPORT, UPDATES AND ONLINE SERVICES

Upon due payment of the agreed License Fees the Licensee will be eligible to receive Support and Updates and to use the Online Services during the License Term, provided, however, that in the event the License Term is longer than 36 months, the initial payment includes Support for only the first 12 months, unless the Parties specifically otherwise agree.

Unless otherwise decided by The Company at its free and absolute discretion, Upgrades will not be included in the Support but may be available subject to additional fees.

From time to time The Qt Company may change the Support terms, provided that during the respective ongoing License Term the level of Support provided by The Qt Company may not be reduced without the consent of the Licensee.

Unless otherwise agreed, The Qt Company shall not be responsible for providing any service or support to Customers.
9. CONFIDENTIALITY

Each Party acknowledges that during the Term of this Agreement each Party may receive information about the other Party's business, business methods, business plans, customers, business relations, technology, and other information, including the terms of this Agreement, that is confidential and of great value to the other Party, and the value of which would be significantly reduced if disclosed to third parties ("Confidential Information"). Accordingly, when a Party (the "Receiving Party") receives Confidential Information from the other Party (the "Disclosing Party"), the Receiving Party shall only disclose such information to employees and Contractors on a need to know basis, and shall cause its employees and employees of its Affiliates to: (i) maintain any and all Confidential Information in confidence; (ii) not disclose the Confidential Information to a third party without the Disclosing Party's prior written approval; and (iii) not, directly or indirectly, use the Confidential Information for any purpose other than for exercising its rights and fulfilling its responsibilities pursuant to this Agreement. Each Party shall take reasonable measures to protect the Confidential Information of the other Party, which measures shall not be less than the measures taken by such Party to protect its own confidential and proprietary information.

Obligation of confidentiality shall not apply to information that (i) is or becomes generally known to the public through no act or omission of the Receiving Party; (ii) was in the Receiving Party's lawful possession prior to the disclosure hereunder and was not subject to limitations on disclosure or use; (iii) is developed independently by employees or Contractors of the Receiving Party or other persons working for the Receiving Party who have not had access to the Confidential Information of the Disclosing Party, as proven by the written records of the Receiving Party; (iv) is lawfully disclosed to the Receiving Party without restrictions, by a third party not under an obligation of confidentiality; or (v) the Receiving Party is legally compelled to disclose, in which case the Receiving Party shall notify the Disclosing Party of such compelled disclosure and assert the privileged and confidential nature of the information and cooperate fully with the Disclosing Party to limit the scope of disclosure and the dissemination of disclosed Confidential Information to the minimum extent necessary.

The obligations under this Section 9 shall continue to remain in force for a period of five (5) years after the last disclosure, and, with respect to trade secrets, for so long as such trade secrets are protected under applicable trade secret laws.

10. FEES, DELIVERY AND PAYMENT

10.1. License Fees

License Fees are described in The Qt Company's standard price list, quote or Purchase Order confirmation or in an appendix hereto, as the case may be.
The License Fees shall not be refunded or claimed as a credit in any event or for any reason whatsoever.

10.2. Ordering Licenses
Licensee may purchase Development Licenses and Distribution Licenses pursuant to agreed pricing terms or, if no specific pricing terms have been agreed upon, at The Qt Company’s standard pricing terms applicable at the time of purchase. Unless specifically otherwise provided, any pricing terms referenced in this Agreement shall be valid for twelve (12) months from the date of this Agreement.

Licensee shall submit all purchase orders for Development Licenses and Distribution Licenses to The Qt Company by email or any other method acceptable to The Qt Company (each such order is referred to herein as a "Purchase Order") for confirmation, whereupon the Purchase Order shall become binding between the Parties.

10.3. Distribution License Packs
Unless otherwise agreed, Distribution Licenses shall be purchased by way of Distribution License Packs.

Upon due payment of the ordered Distribution License Pack(s), the Licensee will have an account of Distribution Licenses available for distributing the Redistributables in accordance with this Agreement.

Each time Licensee distributes a copy of Redistributables, then one Distribution License is used, and Licensee’s account of available Distribution Licenses is decreased accordingly.

Licensee may distribute copies of the Redistributables so long as Licensee has Distribution Licenses remaining on its account.

10.4. Payment Terms
License Fees and any other charges under this Agreement shall be paid by Licensee no later than thirty (30) days from the date of the applicable invoice from The Qt Company.

The Qt Company will submit an invoice to Licensee after the date of this Agreement and/or after The Qt Company receives a Purchase Order from Licensee.

A late payment charge of the lower of (a) one percent per month; or (b) the interest rate stipulated by applicable law, shall be charged on any unpaid balances that remain past due.

10.5. Taxes
All License Fees and other charges payable hereunder are gross amounts but exclusive of any value added tax, use tax, sales tax, withholding tax and
other taxes, duties or tariffs ("Taxes") levied directly for the sale, delivery or use of Licensed Software hereunder pursuant to any applicable law. Such applicable Taxes shall be paid by Licensee to The Qt Company, or, where applicable, in lieu of payment of such Taxes to The Qt Company, Licensee shall provide an exemption certificate to The Qt Company and any applicable authority.

11. RECORD-KEEPING AND REPORTING OBLIGATIONS; AUDIT RIGHTS

11.1. Licensee's Record-keeping
Licensee shall at all times during the Term of this Agreement and for a period of seven (7) years thereafter maintain Licensee's Records in an accurate and up-to-date form. Licensee's Records shall be adequate to reasonably enable The Qt Company to determine Licensee's compliance with the provisions of this Agreement. The records shall conform to general good accounting practices.

Licensee shall, within thirty (30) days from receiving The Qt Company's request to that effect, deliver to The Qt Company a report based on Licensee's Records, such report to contain information, in sufficient detail, on (i) number and identity of users working with Licensed Software or Open Source Qt, (ii) copies of Redistributables distributed by Licensee during the most recent calendar quarter and/or any other term specified by The Qt Company, (iii) number of undistributed copies of Redistributables and corresponding number of unused Distribution Licenses remaining on Licensee's account, and (iv) any other information as The Qt Company may reasonably require from time to time.

11.2. The Qt Company's Audit Rights
The Qt Company or an independent auditor acting on behalf of The Qt Company's, may, upon at least five (5) business days' prior written notice and at its expense, audit Licensee with respect to the Licensee's use of the Licensed Software, but not more frequently than once during each 6-month period. Such audit may be conducted by mail, electronic means or through an in-person visit to Licensee's place of business. Any such in-person audit shall be conducted during regular business hours at Licensee's facilities and shall not unreasonably interfere with Licensee's business activities. The Qt Company or the independent auditor acting on behalf of The Qt Company shall be entitled to inspect Licensee's Records and conduct necessary interviews of Licensee's relevant employees and Contractors. All such Licensee's Records and use thereof shall be subject to an obligation of confidentiality under this Agreement.

If an audit reveals that Licensee is using the Licensed Software beyond scope of the licenses Licensee has paid for, Licensee agrees to pay The Qt Company any amounts owed for such unauthorized use within 30 days from receipt of the corresponding invoice from The Qt Company.
In addition, in the event the audit reveals a material violation of the terms of this Agreement (without limitation, either (i) underpayment of more than 10% of License Fees or 10,000 euros (whichever is more) or (ii) distribution of products, which include or result from Prohibited Combination, shall be deemed a material violation for purposes of this section), then the Licensee shall pay The Qt Company's reasonable cost of conducting such audit.

12. TERM AND TERMINATION

12.1. Agreement Term
This Agreement shall enter into force upon due acceptance by both Parties and remain in force for as long as there is any Development License(s) purchased under this Agreement in force ("Term"), unless and until terminated pursuant to the terms of this Section 12.

12.2. Termination and suspension of rights
Either Party shall have the right to terminate this Agreement upon thirty (30) days prior written notice if the other Party commits a material breach of any obligation of this Agreement and fails to remedy such breach within such notice period.

Instead of termination, The Qt Company shall have the right to suspend or withhold grants of all rights to the Licensed Software hereunder, including but not limited to the Development Licenses, Distribution License, and Support, should Licensee fail to make payment in timely fashion or otherwise violates or is reasonably suspected to violate its obligations or terms of this Agreement, and where such violation or breach is not cured within five (5) business days following The Qt Company's written notice thereof.

12.3. Mutual Right to Terminate
Either Party shall have the right to terminate this Agreement immediately upon written notice in the event that the other Party becomes insolvent, files for any form of bankruptcy, makes any assignment for the benefit of creditors, has a receiver, administrative receiver or officer appointed over the whole or a substantial part of its assets, ceases to conduct business, or an act equivalent to any of the above occurs under the laws of the jurisdiction of the other Party.

12.4. Parties' Rights and Duties upon Termination
Upon expiry or termination of the Agreement, Licensee shall cease and shall cause all Designated Users (including those of its Affiliates' and Contractors') to cease using the Licensed Software and distribution of the Redistributables under this Agreement.

Notwithstanding the above, in the event the Agreement expires or is terminated:
(i) as a result of The Qt Company choosing not to renew the Development License(s) as set forth in Section 3.1, then all valid licenses
possessed by the Licensee at such date shall be extended to be valid in perpetuity under the terms of this Agreement and Licensee is entitled to purchase additional licenses as set forth in Section 10.2; or
(ii) for reason other than by The Qt Company pursuant to item (i) above or pursuant to Section 12.2, then the Licensee is entitled, for a period of six (6) months after the effective date of termination, to continue distribution of Devices under the Distribution Licenses paid but unused at such effective date of termination.

Upon any such termination the Licensee shall destroy or return to The Qt Company all copies of the Licensed Software and all related materials and will certify the same to The Qt Company upon its request, provided however that Licensee may retain and exploit such copies of the Licensed Software as it may reasonably require in providing continued support to Customers.

Expiry or termination of this Agreement for any reason whatsoever shall not relieve Licensee of its obligation to pay any License Fees accrued or payable to The Qt Company prior to the effective date of termination, and Licensee shall immediately pay to The Qt Company all such fees upon the effective date of termination. Termination of this Agreement shall not affect any rights of Customers to continue use of Applications and Devices (and therein incorporated Redistributables).

12.5. Extension in case of bankruptcy
In the event The Qt Company is declared bankrupt under a final, non-cancellable decision by relevant court of law, and this Agreement is not, at the date of expiry of the Development License(s) pursuant to Section 3.1, assigned to party, who has assumed The Qt Company’s position as a legitimate licensor of Licensed Software under this Agreement, then all valid licenses possessed by the Licensee at such date of expiry, and which the Licensee has not notified for expiry, shall be extended to be valid in perpetuity under the terms of this Agreement.

13. GOVERNING LAW AND LEGAL VENUE

In the event this Agreement is in the name of The Qt Company Inc., a Delaware Corporation, then:
(i) this Agreement shall be construed and interpreted in accordance with the laws of the State of California, USA, excluding its choice of law provisions;
(ii) the United Nations Convention on Contracts for the International Sale of Goods will not apply to this Agreement; and
(iii) any dispute, claim or controversy arising out of or relating to this Agreement or the breach, termination, enforcement, interpretation or validity thereof, including the determination of the scope or applicability of this Agreement to arbitrate, shall be determined by arbitration in San Francisco, USA, before one arbitrator. The
arbitration shall be administered by JAMS pursuant to JAMS' Streamlined Arbitration Rules and Procedures. Judgment on the Award may be entered in any court having jurisdiction. This Section shall not preclude parties from seeking provisional remedies in aid of arbitration from a court of appropriate jurisdiction.

In the event this Agreement is in the name of The Qt Company Ltd., a Finnish Company, then:

(i) this Agreement shall be construed and interpreted in accordance with the laws of Finland, excluding its choice of law provisions;
(ii) the United Nations Convention on Contracts for the International Sale of Goods will not apply to this Agreement; and
(iii) any disputes, controversy or claim arising out of or relating to this Agreement, or the breach, termination or validity thereof shall be finally settled by arbitration in accordance with the Arbitration Rules of Finland Chamber of Commerce. The arbitration tribunal shall consist of one (1), or if either Party so requires, of three (3), arbitrators. The award shall be final and binding and enforceable in any court of competent jurisdiction. The arbitration shall be held in Helsinki, Finland and the process shall be conducted in the English language. This Section shall not preclude parties from seeking provisional remedies in aid of arbitration from a court of appropriate jurisdiction.

14. GENERAL PROVISIONS

14.1. No Assignment
Except in the case of a merger or sale of substantially all of its corporate assets, Licensee shall not be entitled to assign or transfer all or any of its rights, benefits and obligations under this Agreement without the prior written consent of The Qt Company, which shall not be unreasonably withheld or delayed. The Qt Company shall be entitled to freely assign or transfer any of its rights, benefits or obligations under this Agreement.

14.2. No Third-Party Representations
Licensee shall make no representations or warranties concerning the Licensed Software on behalf of The Qt Company. Any representation or warranty Licensee makes or purports to make on The Qt Company's behalf shall be void as to The Qt Company.

14.3. Surviving Sections
Any terms and conditions that by their nature or otherwise reasonably should survive termination of this Agreement shall so be deemed to survive. Such sections include especially the following: 1, 2, 6, 7, 9, 11, 12.4, 13 and 14.

14.4. Entire Agreement
This Agreement, the exhibits hereto, the License Certificate and any applicable Purchase Order accepted by The Qt Company constitute the complete
agreement between the Parties and supersedes all prior or contemporaneous discussions, representations, and proposals, written or oral, with respect to the subject matters discussed herein.

In the event of any conflict or inconsistency between this Agreement and any Purchase Order, the terms of this Agreement will prevail over the terms of the Purchase Order with respect to such conflict or inconsistency.

Parties specifically acknowledge and agree that this Agreement prevails over any click-to-accept or similar agreements the Designated Users may need to accept online upon download of the Licensed Software, as may be required by The Qt Company's applicable processes relating to Licensed Software.

14.5. Modifications
No modification of this Agreement shall be effective unless contained in a writing executed by an authorized representative of each Party. No term or condition contained in Licensee's Purchase Order ("Deviating Terms") shall apply unless The Qt Company has expressly agreed such Deviating Terms in writing. Unless and to the extent expressly agreed by The Qt Company, any such Deviating Terms shall be deemed void and with no legal effect. For clarity, delivery of the Licensed Software following the receipt of the Purchase Order including Deviating Terms shall not constitute acceptance of such Deviating Terms.

14.6. Force Majeure
Except for the payment obligations hereunder, neither Party shall be liable to the other for any delay or non-performance of its obligations hereunder in the event and to the extent that such delay or non-performance is due to an event of act of God, terrorist attack or other similar unforeseeable catastrophic event that prevents either Party for fulfilling its obligations under this Agreement and which such Party cannot avoid or circumvent ("Force Majeure Event"). If the Force Majeure Event results in a delay or non-performance of a Party for a period of three (3) months or longer, then either Party shall have the right to terminate this Agreement with immediate effect without any liability (except for the obligations of payment arising prior to the event of Force Majeure) towards the other Party.

14.7. Notices
Any notice given by one Party to the other shall be deemed properly given and deemed received if specifically acknowledged by the receiving Party in writing or when successfully delivered to the recipient by hand, fax, or special courier during normal business hours on a business day to the addresses specified for each Party on the signature page. Each communication and document made or delivered by one Party to the other Party pursuant to this Agreement shall be in the English language.
14.8. Export Control
Licensee acknowledges that the Redistributables, as incorporated in Applications or Devices, may be subject to export control restrictions under the applicable laws of respective countries. Licensee shall fully comply with all applicable export license restrictions and requirements as well as with all laws and regulations relating to the Redistributables and exercise of licenses hereunder and shall procure all necessary governmental authorizations, including without limitation, all necessary licenses, approvals, permissions or consents, where necessary for the re-exportation of the Redistributables, Applications and/or Devices.

14.9. No Implied License
There are no implied licenses or other implied rights granted under this Agreement, and all rights, save for those expressly granted hereunder, shall remain with The Qt Company and its licensors. In addition, no licenses or immunities are granted to the combination of the Licensed Software with any other software or hardware not delivered by The Qt Company under this Agreement.

14.10. Attorney Fees
The prevailing Party in any action to enforce this Agreement shall be entitled to recover its attorney's fees and costs in connection with such action.

14.11. Privacy
Licensee acknowledges and agrees that for the purpose of this Agreement, The Qt Company may collect, use, transfer and disclose personal data pertaining to Designated Users as well as any other employees and directors of the Licensee and its Contractors relevant for carrying out the intent of this Agreement. Such personal data may be collected from the Licensee or directly from the relevant individuals. The Parties acknowledge that with regard to such personal data processed hereunder, The Qt Company shall be regarded as the Data Controller under the applicable Data Protection Legislation. The Qt Company shall process any such personal data in accordance with its privacy policies and practices, which will comply with all applicable requirements of the Data Protection Legislation.

14.12. Severability
If any provision of this Agreement shall be adjudged by any court of competent jurisdiction to be unenforceable or invalid, that provision shall be limited or eliminated to the minimum extent necessary so that this Agreement shall otherwise remain in full force and effect and enforceable.

APPENDICES

The Agreement includes Appendix 1, and possibly one or more of the appendices 3-5, depending on the product(s) purchased by the Licensee, what is stated in the quote or invoice, and/or what is stated in the
Licensee’s License Certificate.

APPENDIX 1

The modules and/or tools that are included in the respective product - Qt for Application Development (QtAD), Qt for Device Creation (QtDC), Qt for MCUs (QtMCU), Qt 3D Studio (Qt3DS) and Qt Design Studio (QtDS) - are marked with ‘X’ in the below table.

Parts of the product that are permitted for distribution in object-code form only (“Redistributables”) are marked with ‘R’ in the below table.

<table>
<thead>
<tr>
<th>Modules / Tools</th>
<th>QtAD</th>
<th>QtDC</th>
<th>QtMCU</th>
<th>Qt3DS</th>
<th>QtDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qt Core</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt GUI</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Multimedia</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Multimedia Widgets</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Network</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt QML</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Controls 2</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Dialogs</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Layouts</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Test</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt SQL</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Test</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Widgets</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active Qt</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt 3D</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Android Extras</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Bluetooth</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Canvas 3D</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Concurrent</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt D-Bus</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Gamepad</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Graphical Effects</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Help</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Image Formats</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Location</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Mac Extras</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Network Authorization</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt NFC</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Platform Headers</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Positioning</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Print Support</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Purchasing</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt for Python</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Controls</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Extras</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Widgets</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt SCXML</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Sensors</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Serial Bus</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Serial Port</td>
<td>X, R</td>
<td>X, R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Speech</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>-------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>------</td>
</tr>
<tr>
<td>Qt SVG</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt UI Tools</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt WebChannel</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt WebEngine</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt WebSockets</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt WebView</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Windows Extras</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt X11 Extras</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt XML</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt XML Patterns</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Wayland Compositor</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Charts</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Data visualization</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Virtual Keyboard</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Boot 2 Qt stack</td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt OTA</td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device Utilities</td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Debugging Bridge (QDB) Daemon</td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Ultralite Controls</td>
<td></td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Ultralite</td>
<td></td>
<td></td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Creator</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Designer (Qt Widget Designer)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Designer (Qt Creator)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>plugin</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Linguist</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Assistant</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Lupdate</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Lrelease</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qmake</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Uic</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Recc</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qlalr</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qdoc</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qmlscene</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qmlviewer</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Target toolchains</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Debugging Bridge (QDB) Host</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>qtconfig-gui</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Emulator</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qmllnterfacegenerator</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qmllntocpp</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>qulfontcompiler</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt53DStudioRuntime2</td>
<td></td>
<td></td>
<td>X, R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt 3D Studio</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Design Studio</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

APPENDIX 3: ADDITIONS TO LICENSED SOFTWARE
In addition to what is provided under the definition of the Licensed Software, Parties agree that Licensed Software shall also include the Add-On Products of The Qt Company, as mentioned in this Appendix, if included in the quote / invoice.

The Modules and/or Tools of the Licensed Software that are included with each Add-On Product respectively are marked with 'X' in the below table.

Parts of the respective Add-On Product that are permitted for distribution in object-code form only ("Redistributables") are marked with 'R' in the below table.

<table>
<thead>
<tr>
<th>Modules / Tools of Licensed Software</th>
<th>Add-On Product(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ Qt M2M</td>
<td>Qt for Automation</td>
</tr>
<tr>
<td>Qt KNX</td>
<td></td>
</tr>
<tr>
<td>Qt OPC UA</td>
<td>Qt OPC UA</td>
</tr>
<tr>
<td>Qt CoAP</td>
<td>Qt CoAP</td>
</tr>
<tr>
<td>Qt Safe Renderer</td>
<td>Qt Safe Renderer</td>
</tr>
<tr>
<td>Qt Application Manager</td>
<td>Qt Application Manager</td>
</tr>
<tr>
<td>Qt IVI</td>
<td>Qt IVI</td>
</tr>
<tr>
<td>Reference UI</td>
<td>Reference UI</td>
</tr>
<tr>
<td>Qt GENIVI Extras</td>
<td>Qt GENIVI Extras</td>
</tr>
<tr>
<td>QML Live</td>
<td>QML Live</td>
</tr>
<tr>
<td>Qt Creator</td>
<td>Qt Creator</td>
</tr>
<tr>
<td>Deployment</td>
<td>Deployment</td>
</tr>
<tr>
<td>Qt Creator Plugin</td>
<td>Qt Creator Plugin</td>
</tr>
<tr>
<td>for Qt Application Manager</td>
<td>for Qt Application Manager</td>
</tr>
<tr>
<td>Qt Automotive Suite</td>
<td>X</td>
</tr>
<tr>
<td>Deployment Server</td>
<td></td>
</tr>
<tr>
<td>Qt Design Studio</td>
<td>X</td>
</tr>
<tr>
<td>Qt 3D Studio</td>
<td>X</td>
</tr>
<tr>
<td>GammaRay</td>
<td>X</td>
</tr>
<tr>
<td>Platform adaptations</td>
<td></td>
</tr>
<tr>
<td>for specified</td>
<td></td>
</tr>
<tr>
<td>Deployment Platforms</td>
<td></td>
</tr>
<tr>
<td>Qt for Device</td>
<td>X</td>
</tr>
<tr>
<td>Creation</td>
<td></td>
</tr>
</tbody>
</table>

All the above Redistributables are subject to applicable provisions and limitations including but not limited to what is defined in section 3 of the Agreement.

APPENDIX 4: SMALL BUSINESS AND STARTUP

The provisions of this Appendix 4 are applicable for companies with an annual revenue, including funding, equivalent to maximum of 250,000 USD (in applicable currency) during the latest full calendar year, as evidenced by duly audited records of the Licensee and approved by The Qt Company ("Start-up Company").

Start-up Companies are qualified for a discounted License Fee for maximum of four (4) Development Licenses ("Start-up Development License") unless otherwise agreed between the parties.

Start-up Development License entitles the respective Designated User for Support only for installation related issues.

Upon expiry of the respective License Term, the Start-up Development Licenses shall be automatically extended, pursuant to Section 3.1 of the Agreement, for a Renewal Term either as new Start-up Development Licenses (if the Licensee still qualifies as a Start-up Company), or as normal Development Licenses (if the Licensee no longer qualifies as a Start-up Company).

APPENDIX 5: NON-COMMERCIAL USE
The provisions of this Appendix 5 are applicable for non-commercial use of the Licensed Software by the Licensee.
For the purpose of this Appendix 5, the following additional definitions (replacing the relevant definition of the Agreement, where applicable) shall be applicable:

"Demo Units" shall mean (i) hardware development platform, which incorporates the Licensed Software along with Licensee's software and/or hardware, and (ii) prototype versions of Applications or Devices.

"Designated User(s)" shall mean the employees and students of the Licensee.

"Licensee Products" shall mean Applications and/or Devices.

"Permitted Purpose" shall mean (i) Licensee's internal evaluation and testing of Licensed Software, (ii) building Demo Units as well as (iii) educational use.

"Term" shall mean a period of twelve (12) months or any such other period as may be agreed between the Parties.

For the purpose of this Appendix 5, the following changes shall be agreed with respect to relevant Sections of the Agreement:

I. Recital (A) shall be replaced in its entirety to read as follows:

"(A) Licensee wishes to use the Licensed Software for the Permitted Purpose."

II. Section 3.1 shall be replaced in its entirety to read as follows:

"The Qt Company grants to Licensee a personal, non-exclusive, non-transferable, revocable, royalty-free license, valid for the Term, to use, modify and copy the Licensed Software solely for the Permitted Purpose.

Licensee may install copies of the Licensed Software on five (5) computers per Designated User, provided that only the Designated Users who have a valid Development License may use the Licensed Software.

Licensee may demonstrate the Demo Units, provided that such demonstrations must be conducted by Licensee, and the Demo Units must remain in Licensee's possession and under Licensee's control at all times.

For clarity, this Agreement does not (i) entitle Licensee to use Licensed Software to create Applications or Devices (other than prototypes thereof) or (ii) carry any distribution rights to Licensee, but such rights are subject to and conditional upon conclusion of a separate license agreement with The Qt Company."

III. Sections 3.2, 3.3, 8 and 10 shall be deleted.

IV. Section 3.4 shall be replaced in its entirety to read as follows:

"Licensee shall not:
- remove or alter any copyright, trademark or other proprietary rights notice contained in any portion of the Licensed Software;
- transfer, publish, sublicense, disclose, display or otherwise
  make the Licensed Software available to any third party (except
  that Licensee may demonstrate the Demo Units pursuant to Section
  3.1);
- in any way combine, incorporate or integrate Licensed Software
  with, or use Licensed Software for creation of, any software
  created with or incorporating Open Source Qt;

Licensee shall cause all Designated Users who make use of the
licenses granted under this Agreement, to be contractually bound
to comply with the relevant terms of this Agreement and not to use
the Licensed Software beyond the terms hereof. Licensee shall be
responsible for any and all actions and omissions of its
Designated Users relating to the Licensed Software and use
thereof.

Any use of Licensed Software beyond the provisions of this
Agreement is strictly prohibited and requires an additional license
from The Qt Company."

V. Section 12 shall be replaced in its entirety to read as follows:
"This Agreement shall enter into force upon due acceptance by both
Parties and remain in force for the Term, unless and until
terminated pursuant to the terms of Section 12.

Upon termination of the Agreement, Licensee shall cease using the
Licensed Software. All other copies of Licensed Software in the
possession or control of Licensee must be erased or destroyed.
An officer of Licensee must, upon request, promptly deliver to
The Qt Company a written confirmation that this has occurred."

Except for the modifications specified above, this Appendix carries no
change to the terms of the Agreement which shall remain in full force.
Copyright (c) 2008-2012, SARL Adaltas.
All rights reserved.

Redistribution and use of this software in source and binary forms, with or
without modification, are permitted provided that the following conditions
are met:

- Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

- Neither the name of SARL Adaltas nor the names of its contributors may be
  used to endorse or promote products derived from this software without
specific prior written permission of SARL Adaltas.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2014-2015 Calvin Metcalf & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Testing Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@google.com>
Ji Sigursson <joj@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@google.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@gmail.com>
Patrick Riley <pfr@google.com>
Peter Kaminski <piotrk@google.com>
Preston Jackson <preston.a.jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@google.com>
Russ Rufer <russ@pentad.com>
Sean Mcafee <eefacm@gmail.com>
Sigurur sgeirsson <siggi@google.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@gmail.com>
Zhanyong Wan <wan@gmail.com>
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

~tools/utils/qtcreator/
Copyright (c) 2013-2018 sha.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 1998 - 2009, Paul Johnston & Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with
We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.
The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.
This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
 prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (C) 2012 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2015, Brandon Jones, Colin MacKenzie IV.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below. This license makes reference to the version 3 of the GNU General Public License, which you can find in the LICENSE.GPLv3 file.

-------------------------------------------------------------------------
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would otherwise
be required to provide such information under section 6 of the
GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the
Combined Work produced by recombining or relinking the
Application with a modified version of the Linked Version. (If
you use option 4d0, the Installation Information must accompany
the Minimal Corresponding Source and Corresponding Application
Code. If you use option 4d1, you must provide the Installation
Information in the manner specified by section 6 of the GNU GPL
for conveying Corresponding Source.)

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Copyright 2018, VideoLAN and dav1d authors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (C) 2016-2019 Erik Barke, Monounity.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Suite 500, Boston, MA 02110-1335 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public...
Licenses are intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages—typically libraries—of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The
form data derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you">

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

^L

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Suite 500, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or
your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Copyright (c) 2011, The WebRTC project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh, and other contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
Open Source Used In DNAC Platform Arica  5966

copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license (Fonts) and associated documentation files (the Font Software), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words Bitstream or the word Vera.

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the Bitstream Vera names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of GNOME, the GNOME Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the GNOME Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.
Copyright (c) 2012, Linux USB Project
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Linux USB Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2014-2016 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2015-2017 Ole Michelsen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2009-2014 Contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

The FreeType Project LICENSE
----------------------------

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine.
various tools and contributions which rely on, or relate to, the
FreeType Project.

This license applies to all files found in such packages, and
which do not fall under their own explicit license. The license
affects thus the FreeType font engine, the test programs,
documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG
(Independent JPEG Group) licenses, which all encourage inclusion
and use of free software in commercial and freeware products
alike. As a consequence, its main points are that:

- We don’t promise that this software works. However, we will be
  interested in any kind of bug reports. (`as is' distribution)

- You can use this software for whatever you want, in parts or
  full form, without having to pay us. (`royalty-free' usage)

- You may not pretend that you wrote this software. If you use
  it, or only parts of it, in a program, you must acknowledge
  somewhere in your documentation that you have used the
  FreeType code. (`credits')

We specifically permit and encourage the inclusion of this
software, with or without modifications, in commercial products.
We disclaim all warranties covering The FreeType Project and
assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a
credit/disclaimer to use in compliance with this license. We thus
encourage you to use the following text:

""
Portions of this software are copyright <year> The FreeType
Project (www.freetype.org). All rights reserved.
""

Please replace <year> with the value from the FreeType version you
actually use.

Legal Terms
============

0. Definitions
-------------
Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

---------------

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

---------------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying
documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising
-------------

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: 'FreeType Project', 'FreeType Engine', 'FreeType library', or 'FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts
-----------

There are two mailing lists related to FreeType:

- freetype@nongnu.org

  Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

- freetype-devel@nongnu.org
Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

https://www.freetype.org

--- end of FTL.TXT ---

Copyright (C) 2000 TurboLinux, Inc. Written by Justin Yu and Sean Chen.
Copyright (C) 2001, 2002 Turbolinux, Inc. Written by James Su.
Copyright (C) 2001, 2002 ThizLinux Laboratory Ltd. Written by Anthony Fok.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
## Change Log

### v25.0.1 (2019/01/10 19:29 +00:00)
- [f1c835b](https://github.com/davglass/license-checker/commit/f1c835b991b65dc928eae64c29b73df9e703169) Updated readme with new args (@davglass)
- [ada667f](https://github.com/davglass/license-checker/commit/ada667f28e8ef4e96b5b6059fe05541d05ca936c) changelog (@davglass)
- [7813f0d](https://github.com/davglass/license-checker/commit/7813f0d3b351f7cc4cd37d8e7b4670f9005fed06) contrib (@davglass)

### v25.0.0 (2019/01/10 15:09 +00:00)
- [cdb77d9](https://github.com/davglass/license-checker/commit/cdb77d9a2b36cd87503f32265a0ee388b25c24ae) 25.0.0 (@davglass)
- [b897345](https://github.com/davglass/license-checker/commit/b89734549bc05fb5e143083df6d3959d87eb2b8) [Fixes #136] Document --start option and add the depth option for looking into root dependencies or infinitely (@davglass)
- [#185](https://github.com/davglass/license-checker/pull/185) Merge pull request #185 from theboolean/exclude-packages-and-private (@theboolean)
- [#184](https://github.com/davglass/license-checker/pull/184) Merge pull request #184 from theboolean/docs/fix-typo (@theboolean)
- [987a60f](https://github.com/davglass/license-checker/commit/987a60fbb58052cf36d5a80e6252e81366b1dd8) changes 24.1.0 (@davglass)

### v24.1.0 (2018/12/14 20:54 +00:00)
- [18ab8a1](https://github.com/davglass/license-checker/commit/18ab8a1eb478d1354429ca4318c1df3de3e4cb1) 24.1.0 (@davglass)
- [1dca0f6](https://github.com/davglass/license-checker/commit/1dca0f616aebdef2d398c130a2a4f86c2b4e88c8) Changelog (@davglass)
- [#188](https://github.com/davglass/license-checker/pull/188) Fix for #153: remove deprecated package 'spdx' (#188) (@rogierschouten)
- [d1d34b1](https://github.com/davglass/license-checker/commit/d1d34b1a8894987807419c093c3ec10b4f06db3d) Fix typo in package.json description (@theboolean)
- [f5c9008](https://github.com/davglass/license-checker/commit/f5c9008ab620bb03e21c0d8130e52be526a5bfe8) Allow excluding packages and private packages together (@theboolean)
- [98c4dc6](https://github.com/davglass/license-checker/commit/98c4dc6760fdebb6e58b6238ebcb12302e5b1db5) changes (@davglass)

### v24.0.1 (2018/10/11 16:40 +00:00)
- [d72ae8](https://github.com/davglass/license-checker/commit/d72ae88cb8bd8562e6d12c5e2664e9c19135709) 24.0.1 (@davglass)
- [9433e64](https://github.com/davglass/license-checker/commit/9433e640ea9f629aad5f00c224a9f61026b8116d) [Fixes #181] - Removed extra line break with --cvs (@davglass)
- [65b9a28](https://github.com/davglass/license-checker/commit/65b9a28f532f87e7bb75a32578c7ac5837706d12f) contrib and changes (@davglass)

### v24.0.0 (2018/10/09 21:12 +00:00)
- [84d64a2](https://github.com/davglass/license-checker/commit/84d64a28fa971c3bbf20b9e9de27cafbe6e8481b) 24.0.0 (@davglass)
- [#180](https://github.com/davglass/license-checker/pull/180) Feature: Exclude Private Packages (#180) (Jonny Reeves)
- [#179](https://github.com/davglass/license-checker/pull/179) Merge pull request #179 from jonny-improbable/fix/fix-strict-checks (@jonny-improbable)
- [179077d](https://github.com/davglass/license-checker/commit/179077d079acbcd0183aa6691cfe5c2ec0f93f07) Fixes 176, Packages are not ignored when using --onlyAllow and --excludePackages. (@jonny-improbable)
- [890ca17](https://github.com/davglass/license-checker/commit/890ca17d179bc43a6cb7cf962d69d1521dc9ac2) changelog (@davglass)

### v23.0.0 (2018/10/09 14:37 +00:00)
- #243287](https://github.com/davglass/license-checker/commit/f243287fbce6da33838a226ddf8e412535390d4) 23.0.0 (@davglass)
- [#178](https://github.com/davglass/license-checker/pull/178) Merge pull request #178 from jonny-improbable/fix/filtering (@jonny-improbable)
- [#175](https://github.com/davglass/license-checker/pull/175) Merge pull request #175 from jonny-improbable/patch-1 (@jonny-improbable)
- [2c6611](https://github.com/davglass/license-checker/commit/f2c6611d6c68e50b2b6f56260026837e7167f44a) Fix #177, --packages and --excludePackages filtering (@jonny-improbable)
- [a598df](https://github.com/davglass/license-checker/commit/0a598df55c3ec33803ef33be07cc47a9100ee961) Improve --onlyAllow violation Error
- [ee723af](https://github.com/davglass/license-checker/commit/ee723af79889a4d98ed067494a58f6aa41a44ac7) changes (@davglass)

### v22.0.0 (2018/10/08 19:00 +00:00)
- bd398c3](https://github.com/davglass/license-checker/commit/bd398c3bce0279699c9aad1dc4b592f398a75766) 22.0.0 (@davglass)
- [#172](https://github.com/davglass/license-checker/pull/172) Merge pull request #172 from fzaninotto/exclude-packages (@fzaninotto)
- [014fd96](https://github.com/davglass/license-checker/commit/014fd9613fe5460e0b7a0fc118a659a337e25e2c) added changes (@davglass)

### v21.0.0 (2018/07/30 13:49 +00:00)
- d75fe43](https://github.com/davglass/license-checker/commit/d75fe43d61a9d5b6696d71a663e25a53f3d44f7) 21.0.0 (@davglass)
- [05cb221](https://github.com/davglass/license-checker/commit/05cb221359de4ffe1039f586cd9e8ff7cb9d6398) [Fixes #174] - Moved private and unknown detection above filtering (@davglass)
- [b1d5700](https://github.com/davglass/license-checker/commit/b1d570088236dfb90253bdac27763523b2e74ff6) Add an excludePackages option (@fzaninotto)
- [231bd3f](https://github.com/davglass/license-checker/commit/231bd3f2412507be0d20a10b23c1b1fe93602) updated readme (@davglass)
- [6c61a7b](https://github.com/davglass/license-checker/commit/6c61a7bd102466c702065d42706392c11717cc1d) changelog (@davglass)

### v20.2.0 (2018/07/30 13:49 +00:00)
- [28dc521](https://github.com/davglass/license-checker/commit/28dc5218dec3340778bd801593275ad1b222e53f) 20.2.0 (@davglass)
- [#169](https://github.com/davglass/license-checker/pull/169) Merge pull request #169 from codepunkt/restrict_packages (@codepunkt)
- [5e4b274](https://github.com/davglass/license-checker/commit/5e4b274e955485fb2aba37e65fee913b8042f38d) refactoring: restore previous code style (@codepunkt)
- [0f53922](https://github.com/davglass/license-checker/commit/0f539222088be0b0f94f8db7ea884b53d0137e45) feat: allow restricting output to specified packages (@codeunkt)
- [5f63f87](https://github.com/davglass/license-checker/commit/5f63f8766a17e589b57db99bda4eb54d91b85b77) changes and contrib (@davglass)

### v20.1.0 (2018/06/07 14:33 +00:00)
- [970882a](https://github.com/davglass/license-checker/commit/970882a90b345c7f32496431e5ca4c069ad7e320) 20.1.0 (@davglass)
- [#165](https://github.com/davglass/license-checker/pull/165) Merge pull request #165 from santiagocanti/fix-exclude-not-adding-custom-licenses (@santiagocanti)
- [89e48a9](https://github.com/davglass/license-checker/commit/89e48a9d30c9f248bfce0dc3563ce4b652e22) Fix exclude not including custom licenses (@santiagocanti)

### v20.0.0 (2018/05/17 14:14 +00:00)
- [0e89310](https://github.com/davglass/license-checker/commit/0e8931021df777cb401d9a332fa2a8c9a2c49eb2) 20.0.0 (@davglass)
- [c36a5c9](https://github.com/davglass/license-checker/commit/c36a5c99bb4f688164e19cad5090ba3c7ba67a) changes (@davglass)
- [ccb66f3](https://github.com/davglass/license-checker/commit/ccb66f3410a4589e6f2b7cf13ae0f1446bab46) contributors (@davglass)
- [#156](https://github.com/davglass/license-checker/pull/156) Merge pull request #156 from zodiac403/z4/copyright (@zodiac403)
- [56414b4](https://github.com/davglass/license-checker/commit/56414b4a67cbb25bcc8eb8cc05b0ac712b677dc2) Add unit test for Copyright statement. (@zodiac403)
- [3eafe24](https://github.com/davglass/license-checker/commit/3eafe249ad662af727a9b7472f7ad6c1b9a73ade) fixing test coverage with switch to nyc from istanbul (@davglass)

### v19.0.0 (2018/04/30 14:51 +00:00)
- [7e83a06](https://github.com/davglass/license-checker/commit/7e83a06d23c104c3d8257eb64666704ecca465) 19.0.0 (@davglass)
- [e5eb49b](https://github.com/davglass/license-checker/commit/e5eb49bb5546db681736f159d63908cd39ce9) upgrading packages (@davglass)
- [442ae6](https://github.com/davglass/license-checker/commit/442ae69b5d44e47bf2e441c4e1f86d387e4f6c) changelog (@davglass)

### v18.1.0 (2018/04/30 14:16 +00:00)
- [82ec91b](https://github.com/davglass/license-checker/commit/82ec91bb02f6b9f8164b62b4a827198156ad4c) 18.1.0 (@davglass)
- [4b546d3](https://github.com/davglass/license-checker/commit/4b546d38ce91bcb8d217bc53e44635b0444f2) contrib (@davglass)
- [#155](https://github.com/davglass/license-checker/pull/155) Merge pull request #155 from ktmouk/supress-path-property (@ktmouk)
- [#160](https://github.com/davglass/license-checker/pull/160) Merge pull request #160 from helio-frota/pr (@helio-frota)
- [8b068f2](https://github.com/davglass/license-checker/commit/8b068f25fe910d087c06c29312bc250b49aba4c7) fix: Updating debug dependency to fix a security warning (@helio-frota)
- [040dd94](https://github.com/davglass/license-checker/commit/040dd9468261966c5c40e4a418aa9ba8f716c3a2) Copyright statement: Handle multi-line staments (@zodiac403)
- [0480750](https://github.com/davglass/license-checker/commit/04807506d61aa1d704e8b51da3a289baa025b510) Fix layout issue. (@zodiac403)
- [9cb45f2](https://github.com/davglass/license-checker/commit/9cb45f2b9ce2c1142e0d1f556f01f7d754737a4c) Link with customFormat (@zodiac403)
- [f83aaaee](https://github.com/davglass/license-checker/commit/f83aaaeeb84ad4cc2e9880802951dfacbc8660c2d7f) Merge branch 'z4/copyright' of https://github.com/davglass/license-checker into z4/copyright (@zodiac403)
- [7536af9](https://github.com/davglass/license-checker/commit/7536af935d91078657f0e169ed82383f99607f6) Trim output. (@zodiac403)
- [8716845](https://github.com/davglass/license-checker/commit/87168456ca35263882bd2b2b66d0081f2f8b6b66) Suppress path property (@ktmouk)
- [be57ee8](https://github.com/davglass/license-checker/commit/be57ee8f76c8aad0d227913b70e12ac0dc640) Add example for (c) statement. (@zodiac403)
- [fe17f7c](https://github.com/davglass/license-checker/commit/fe17f7c76dbdae0cbf689a0cd260a746668f3281) Parse (c) statements from LICENSE file. (@zodiac403)
- [#149](https://github.com/davglass/license-checker/pull/149) Merge pull request #149 from honzajavorek/patch-1 (@honzajavorek)
- [955bbf6](https://github.com/davglass/license-checker/commit/955bbf60f70ffcd9f0d01ef37bb84fbbac65e007b) remove the old status badge (@honzajavorek)

### v18.0.0 (2018/03/20 12:30 +00:00)
- [315de0b](https://github.com/davglass/license-checker/commit/315de0b1a811cf8737c3c0131c217c4820df13a8) 18.0.0 (@davglass)
- [76655d5](https://github.com/davglass/license-checker/commit/76655d59bad95c2e52724aa3a0cf036f0e608d6) changelog (@davglass)
- [c8aa823](https://github.com/davglass/license-checker/commit/c8aa82342b7a250d510582c8a03dd76af50bd9c) added contributors (@davglass)
- [#147](https://github.com/davglass/license-checker/pull/147) Merge pull request #147 from honzajavorek/honzajavorek/license-files (@honzajavorek)
- [#148](https://github.com/davglass/license-checker/pull/148) Merge pull request #148 from honzajavorek/honzajavorek/use-stderr (@honzajavorek)
- [#146](https://github.com/davglass/license-checker/pull/146) Merge pull request #146 from honzajavorek/honzajavorek/fix-e2e-test (@honzajavorek)
- [#145](https://github.com/davglass/license-checker/pull/145) Merge pull request #145 from honzajavorek/honzajavorek/drop-4-add-8 (@honzajavorek)
- [0105f7d](https://github.com/davglass/license-checker/commit/0105f7d01a67b877c43471d95a1d76b191d5ed) use stderr where applicable (@honzajavorek)
- [0ff8d91](https://github.com/davglass/license-checker/commit/0ff8d917659446256074c0940ae43dd84930233) add support for LICENSE-MIT and similar (@honzajavorek)
- [02589e](https://github.com/davglass/license-checker/commit/02589e62c0ba8221081b881799d561c3eff20500) isolate and test the license file detection (@honzajavorek)
- [0da0fe](https://github.com/davglass/license-checker/commit/0da0fe2ab28f1658e4ed504bb034fb1536b9f8) add warning about commas and test it (@honzajavorek)
- [3db0b6f](https://github.com/davglass/license-checker/commit/3db0b6f05a68ccff4181503ca87edece4ef10cd7) fix e2e test (transition from comma delimiter to semicolon) (@honzajavorek)
- [ce3b73](https://github.com/davglass/license-checker/commit/ce3b73505ac40a2dca211e78427624d009585d) let Travis CI to cache node_modules (@honzajavorek)
- [8cea3dc](https://github.com/davglass/license-checker/commit/8ceaa3dcecf3d185dc3ae3f37f44a8d7e123a1988) add Travis CI badge to the README (@honzajavorek)
- [8f59c3b](https://github.com/davglass/license-checker/commit/8f59c3b94f69b1d51535bceff80c0e0060d6ef6b) drop support for node 4, add support for node 8 (@honzajavorek)

### v17.0.0 (2018/03/19 13:26 +00:00)
- [cd5fdd8](https://github.com/davglass/license-checker/commit/cd5fdd8e94267d78f29e86dd9cf8b694c037f) 17.0.0 (@davglass)
- [13c11ee](https://github.com/davglass/license-checker/commit/13c11ee8a1b1a3e012de6253600fce91c355c8b) updated readme and help text (@davglass)
- [#144](https://github.com/davglass/license-checker/pull/144) Merge pull request #144 from creising/parser-fixes (@creising)
- [d2ab8df](https://github.com/davglass/license-checker/commit/d2ab8df30bc3eb38b321a12a70cf477a69e0) added support for lic with commas and spaces (@creising)
- [2075208](https://github.com/davglass/license-checker/commit/20752083c6029823f389a30a1ed4d7c90fa927) contrib and changelog (@davglass)
- [6c5274c](https://github.com/davglass/license-checker/commit/6c5274c60d4d318844cf36158da83e0893f08) fixes #132 - Fixed issue with asFiles and path issues (@davglass)
- [a46e448](https://github.com/davglass/license-checker/commit/a46e44838723d9498a56b7ae3cd68164de0a) 16.0.0 (@davglass)
- [38d63e2](https://github.com/davglass/license-checker/commit/38d63e2c2c3fd6c11012bf69b1470b2a06d5554) fixes #128 - Added --onlyAllow option (@davglass)

### v16.0.0 (2018/01/18 15:02 +00:00)
- [#131](https://github.com/davglass/license-checker/pull/131) Merge pull request #131 from mischah/patch-1 (@mischah)
- [e341d17](https://github.com/davglass/license-checker/commit/e341d1701092be87309ec4b51fa6eb23f7e9a589) Fix typo in Readme (@mischah)
- [cc38a20](https://github.com/davglass/license-checker/commit/cc38a2047bf41b6fe2526614a13a9bf54133e04) updated license text (@davglass)

### v15.0.0 (2017/10/19 14:24 +00:00)
- [a9bd001](https://github.com/davglass/license-checker/commit/a9bd001b649bfa186b6a5598b8c67ff56383599) 15.0.0 (@davglass)
- [db621fa](https://github.com/davglass/license-checker/commit/db621f6beebff870699c33be484bf49816ab7866) changelog (@davglass)
- [#124](https://github.com/davglass/license-checker/pull/124) Merge pull request #124 from neverendingqs/app/stricterguesses (@neverendingqs)
- [#126](https://github.com/davglass/license-checker/pull/126) app - output now includes path to dependency. (@neverendingqs)
- [9f68b29](https://github.com/davglass/license-checker/commit/9f68b29b7e3278cdab92e221b6ccbab191932) app - strengthening license guessing; resolves #123. (@neverendingqs)
- [3061e87](https://github.com/davglass/license-checker/commit/3061e37363f3436d86ca6c2ba7b2bfa77def0e73) changes (@davglass)

### v14.0.0 (2017/08/28 13:22 +00:00)
Open Source Used In DNAC Platform Arica 5981

- [b2a4403](https://github.com/davglass/license-checker/commit/b2a440309c1537c52e5c5a2106f11e72d5b784ec) 14.0.0 (@davglass)
- [#122](https://github.com/davglass/license-checker/pull/122) Added --failOn [list] mode (#122) (@tobilg)

### v13.1.0 (2017/08/25 15:00 +00:00)
- [ee25265](https://github.com/davglass/license-checker/commit/ee25265f92fe0d3b603da6b8c88d175e0d1f2f8a5) 13.1.0 (@davglass)
- [5bb9a17](https://github.com/davglass/license-checker/commit/5bb9a17f89d7e180b3a9d929ebc073135262bd2f) changes (@davglass)
- [b1e0392](https://github.com/davglass/license-checker/commit/b1e0392afa4b511b7c532125cd3165a82d3d4e3) added contrib (@davglass)
- [#121](https://github.com/davglass/license-checker/pull/121) Add `--summary` mode to show a summary with counts of the used licenses (#121) (@tobilg)
- [#119](https://github.com/davglass/license-checker/pull/119) Fix minor spelling mistakes (#119) (@tobiasbueschel)
- [#118](https://github.com/davglass/license-checker/pull/118) Minor readme text improvements (#118) (@peteruithoven)
- [d2bc5ea](https://github.com/davglass/license-checker/commit/d2bc5ea91ab50bece91d5f0f050955fe9a6ac59) added license parsing info to README (@davglass)
- [3e9d320](https://github.com/davglass/license-checker/commit/3e9d3200ac2e9f142976da4355680f34a9b0609d) changes (@davglass)

### v13.0.3 (2017/08/04 17:15 +00:00)
- [2caf76b](https://github.com/davglass/license-checker/commit/2caf76be7f2849e153ee28410f3b7a41df3283b5) 13.0.3 (@davglass)
- [fadddc8](https://github.com/davglass/license-checker/commit/fadddc895ef9c6f182e2283c5605a8c1d8fa8a25) [fixes #117] fixed minor URL parsing (@davglass)
- [e58a868](https://github.com/davglass/license-checker/commit/e58a868268ae9d735b364bd87d03140d76482) changes (@davglass)

### v13.0.2 (2017/07/18 16:24 +00:00)
- [2c712b7](https://github.com/davglass/license-checker/commit/2c712b7c7e6efe916d60f3ef3125f7f019c46d59) 13.0.2 (@davglass)
- [ef812c1](https://github.com/davglass/license-checker/commit/ef812c1168901d08b97db450a64ca03bb7062ad) [fixes #114] - Custom URL/File parsing skipped if file found on disk (@davglass)

### v13.0.1 (2017/07/17 12:48 +00:00)
- [a1bf8f1](https://github.com/davglass/license-checker/commit/a1bf8f152b100d81b44757b3b7e141d6fa971c44) 13.0.1 (@davglass)
- [e6d0c89](https://github.com/davglass/license-checker/commit/e6d0c893b8a86e22493f7a5e22fd74ef26267994) contrib & changes (@davglass)
- [#113](https://github.com/davglass/license-checker/pull/113) Use LICENSE file in preference to README (#113) (@ibeucaly)
- [6b7f79f](https://github.com/davglass/license-checker/commit/6b7f79f654e21903ea37e4391fba3601ece32e90) changes (@davglass)

### v13.0.0 (2017/07/14 12:49 +00:00)
- [dd9e880](https://github.com/davglass/license-checker/commit/dd9e880abd234d683d8c1e57e7a5f889e9156de4)
13.0.0 (@davglass)
- [051b20b](https://github.com/davglass/license-checker/commit/051b20b0c4b35f223f724705a0025dafc809373f) cleaned up some tests (@davglass)
- [#110](https://github.com/davglass/license-checker/pull/110) FEATURE: Recognise some custom licenses by file or url (#110) (@dancrumb)
- [#111](https://github.com/davglass/license-checker/pull/111) Resolves #60 (#111) Adding support for GPL and LGPL (@dancrumb)
- [#109](https://github.com/davglass/license-checker/pull/109) FEATURE: Improve license exclusion (#109) (@dancrumb)
- [#108](https://github.com/davglass/license-checker/pull/108) FEATURE: Detect private modules and report them as UNLICENSED (#108) (@dancrumb)

### v12.1.0 (2017/07/12 16:48 +00:00)
- [dc944e5](https://github.com/davglass/license-checker/commit/dc944e54d55d77b0ef6b6321d9abee847662cc8) 12.1.0 (@davglass)
- [d415c58](https://github.com/davglass/license-checker/commit/d415c580b04bc5eea5f92dc9f9aa31b244c1bd3) updated contrib and changes (@davglass)
- [#107](https://github.com/davglass/license-checker/pull/107) Merge pull request #107 from dancrumb/feature/support-for-public-domain-licenses (@dancrumb)
- [0c55770](https://github.com/davglass/license-checker/commit/0c55770294167161c48b798e2a15a545154a87) FEATURE: Add support for Public Domain licenses (@dancrumb)

### v12.0.0 (2017/07/05 13:33 +00:00)
- [8b74fa1](https://github.com/davglass/license-checker/commit/8b74fa1f43b5ed8d217aa3ce86f4662eb024621) 12.0.0 (@davglass)
- [4f3367c](https://github.com/davglass/license-checker/commit/4f3367c2b2e9713af6cccd1134f7f4fa70ed400) changes & contrib (@davglass)
- [#106](https://github.com/davglass/license-checker/pull/106) Merge pull request #106 from helio-frota/licensefile-readme (@helio-frota)
- [a229507](https://github.com/davglass/license-checker/commit/a2295073d0178dca1b88e85ea8f52c21d3be52fc) src: the README is added as licenseFile (@helio-frota)
- [b59d90d](https://github.com/davglass/license-checker/commit/b59d90d8705f6f728c81789936f95097d26857b) changes & contrib (@davglass)

### v11.0.0 (2017/05/30 13:33 +00:00)
- [258df10](https://github.com/davglass/license-checker/commit/258df102da484943088da8e0b6160b80c6e703c8d) 11.0.0 (@davglass)
- [#102](https://github.com/davglass/license-checker/pull/102) Merge pull request #102 from xswordsx/multiple-licenses (@xswordsx)
- [c5c8b0f](https://github.com/davglass/license-checker/commit/c5c8b0f408f2a26259b7b1109f7e335ed5bf9e4) Check for SPDX-compatible licenses (@xswordsx)

### v10.1.0 (2017/05/25 16:16 +00:00)
- [b52a16d](https://github.com/davglass/license-checker/commit/b52a16d0ba414e7287f82325dd65619cd7eafdd1) 10.1.0 (@davglass)
- [#911a99](https://github.com/davglass/license-checker/pull/911a9919033254945beb69362cb171436688e) contrib & changes (@davglass)
- [#100](https://github.com/davglass/license-checker/pull/100) Suppress JSON properties with custom format
Open Source Used In DNAC Platform Arica  5983

- [ed5b3ab](https://github.com/davglass/license-checker/commit/ed5b3ab50825f0be032958f8233dd3000ee98e85) changelog (@davglass)

### v10.0.0 (2017/05/17 13:51 +00:00)
- [b2c2283](https://github.com/davglass/license-checker/commit/b2c2283cf5a937e64a07ab200ec2379e481903b6) 10.0.0 (@davglass)
- [a0406ac](https://github.com/davglass/license-checker/commit/a0406ac2c7a0b1037ca31b47e36d3169c088938) [fixes #96] - do not replace newline, etc in licenses for exporting to json/markdown (@davglass)
- [98a03fa](https://github.com/davglass/license-checker/commit/98a03fa700ef6d038ccf91904ea5cf17c82a10ad) [fixes #97] - Added a newline to the output file (@davglass)
- [#98](https://github.com/davglass/license-checker/pull/98) Merge pull request #98 from davglass/eslint (@davglass)
- [10fe441](https://github.com/davglass/license-checker/commit/10fe441a2e70e549e57f7da99d2287fc53555500) moved to eslint (@davglass)
- [f82b535](https://github.com/davglass/license-checker/commit/f82b535d3b7f152fa00bb4680e94af18d1c1421d) changelog (@davglass)

### v9.0.3 (2017/04/10 13:42 +00:00)
- [af0fd60](https://github.com/davglass/license-checker/commit/af0fd60f972b7c84a32117b3dfc1f95bfa2282f27) 9.0.3 (@davglass)
- [#95](https://github.com/davglass/license-checker/pull/95) Merge pull request #95 from abtris/prefix-component (@abtris)
- [3b80272](https://github.com/davglass/license-checker/commit/3b80272a217687c2f0cd7a0c1e98775064d820ff) test: add test for csvComponentPrefix (@abtris)
- [593554](https://github.com/davglass/license-checker/commit/d593554e1a6bf47e004e4a23fca88cb7294f8351) feat: add support for csvComponentPrefix (@abtris)
- [5b2eea1](https://github.com/davglass/license-checker/commit/5b2eea12d8c54068a01a2cf1fd9bdec8e1f0fa2) added contrib (@davglass)

### v9.0.2 (2017/04/05 14:55 +00:00)
- [f24833b](https://github.com/davglass/license-checker/commit/f24833bcee846f608ec3d8c327d335da56c42b895) 9.0.2 (@davglass)
- [#94](https://github.com/davglass/license-checker/pull/94) Merge pull request #94 from timbru31/fixed-comma-exclude (@timbru31)
- [2d08a77](https://github.com/davglass/license-checker/commit/2d08a7700303627ab03ac12358dd524c0be47eee) contrib & changelog (@davglass)

### v9.0.1 (2017/04/05 13:09 +00:00)
- [9f1a1f3](https://github.com/davglass/license-checker/commit/9f1a1f32194a9bb312858f624e303af1ad4f61ca) 9.0.1 (@davglass)
- [c99f612](https://github.com/davglass/license-checker/commit/c99f612fe22d9eb03db0ad41e7918fba6f6e8ed9) added test for licenseText (@davglass)
- [#93](https://github.com/davglass/license-checker/pull/93) Merge pull request #93 from abtris/licenseText-just-in-customFormat (@abtris)
- [8d7fb2f](https://github.com/davglass/license-checker/commit/8d7fb2f855cf585d8e75ab1459e0b0cef43993ba4) fix: allow escaped commas in exclude string (@timbru31)
- [da4d73c](https://github.com/davglass/license-checker/commit/da4d73c0e222f500d6bce54364eef5b7bcc7f1a4)
test: fix test, remove check from non custom format (@abtris)
- [dcd3b59](https://github.com/davglass/license-checker/commit/dcd3b59bb9457df354fa3095189235b6e030df1a)
  feat: add licenseText only for customFormat
- [4e02eca](https://github.com/davglass/license-checker/commit/4e02eca6e8bd332ff6d90e5039caeb50f0df1bf3)
  updated contribs and changelog (@davglass)

### v9.0.0 (2017/04/04 12:26 +00:00)
- [267c80e](https://github.com/davglass/license-checker/commit/267c80efac89eef70c71a12b8f12b3dc8e6d58dd)
  9.0.0 (@davglass)
- [057af47](https://github.com/davglass/license-checker/commit/057af47bc52ac526f646ba21496f338689801809)
  added licenseFile assertion to tests (@davglass)
- [#92](https://github.com/davglass/license-checker/pull/92)
  Merge pull request #92 from abtris/includeFileContent (@abtris)
- [1cda118](https://github.com/davglass/license-checker/commit/1cda118e1aa18cc391ba2765dcaee89172943a0f)
  fix: replace problematics chars for easy import to Excel (@abtris)
- [dd6ecab](https://github.com/davglass/license-checker/commit/dd6ecab1a7b1354c4d1d00149f15db0f5e3e63)
  feat: add licenseText (@abtris)
- [6395498](https://github.com/davglass/license-checker/commit/639549890c54a12043244579ad0c0d5972309f2)
  changes and contrib (@davglass)

### v8.0.4 (2017/02/15 16:21 +00:00)
- [28d379a](https://github.com/davglass/license-checker/commit/28d379aca51da7a47af324bc6feb63736ea02295)
  8.0.4 (@davglass)
- [#89](https://github.com/davglass/license-checker/pull/89)
  Merge pull request #89 from Pawamoy/master (@Pawamoy)
- [#90](https://github.com/davglass/license-checker/pull/90)
  Merge pull request #90 from drewfish/bsd-source-code (@drewfish)
- [b704b1](https://github.com/davglass/license-checker/commit/b704b18a43de246302b8ff6933f8fe5f96ee3a6)
  guess BSD-Source-Code license (@drewfish)
- [8d31eab](https://github.com/davglass/license-checker/commit/8d31eabf6b671d32e6f3e63215d6530dfb4acc59)
  Add docs about customFormat option (@pawamoy)
- [2529d31](https://github.com/davglass/license-checker/commit/2529d3118dfae53f1e01056ec99f5c542cb8d63e)
  changes (@davglass)

### v8.0.3 (2016/11/10 17:04 +00:00)
- [66fd7b5](https://github.com/davglass/license-checker/commit/66fd7b5ad5577cc2c85668a7c6f07365c86a2c85)
  8.0.3 (@davglass)
- [#87](https://github.com/davglass/license-checker/pull/87)
  Merge pull request #87 from drewfish/guess-cc0 (@drewfish)
- [668bbd3](https://github.com/davglass/license-checker/commit/668bbd38fd5148f99c5263f4e0506e04d5e3f8a0)
  guess CC0-1.0, upgraded jshint (@drewfish)

### v8.0.2 (2016/11/09 21:57 +00:00)
- [0bdeedd8](https://github.com/davglass/license-checker/commit/0bdeedd8cb990e1f5ffacde1c12cc708185a5b866)
  8.0.2 (@davglass)
- [ef67bb4](https://github.com/davglass/license-checker/commit/ef67bb454373352343faa9d4de978a295bd1544)
  updated semver (@davglass)
- [14b5c85](https://github.com/davglass/license-checker/commit/14b5c8551cebb5c5ce8aa76fb5367857ce891d013)
Changelog (@davglass)

### v8.0.1 (2016/11/09 21:51 +00:00)
- [594871d](https://github.com/davglass/license-checker/commit/594871d6197a030121f940500495e6ca7d82f02c) 8.0.1 (@davglass)
- [c243663](https://github.com/davglass/license-checker/commit/c243663b0ab8a0126251292a57b2c47369a51eef) added contrib (@davglass)
- [#86](https://github.com/davglass/license-checker/pull/86) Merge pull request #86 from drewfish/copying (@drewfish)
- [380aa26](https://github.com/davglass/license-checker/commit/380aa26799cb527739907484e69c6dae26d5f17a) also support COPYING file (@drewfish)
- [329c367](https://github.com/davglass/license-checker/commit/329c3676d0cd5c5947a53fd205ff23b767b7f882) handling undefined@undefined error better (@davglass)
- [1f93a07](https://github.com/davglass/license-checker/commit/1f93a072632aad0660b63f61c56c3bd2fe109474) changelog (@davglass)

### v8.0.0 (2016/09/09 13:23 +00:00)
- [44c2c48](https://github.com/davglass/license-checker/commit/44c2c48ae177915523fd89a74474fe0953408bf) 8.0.0 (@davglass)
- [b5f5c6a](https://github.com/davglass/license-checker/commit/b5f5c6aa03a86c60c52fd308f71721744a4bbaaa99) [fixes #78] - include notice file if found (@davglass)
- [dfcbd6f](https://github.com/davglass/license-checker/commit/dfcbd6f49f66ac74e065f71e45cb08ba8fe62ab) changelog (@davglass)
- [99976d0](https://github.com/davglass/license-checker/commit/99976d07980e241384be10341d0305d08ede9cea5) 7.1.1 (@davglass)
- [ef857b7](https://github.com/davglass/license-checker/commit/ef857b7e9bab706299a1cc565ec5d144893af424) [fixes #77] fixed readme file name parsing, made it more strict (@davglass)

### v7.1.1 (2016/09/07 19:16 +00:00)
- [28be6d4](https://github.com/davglass/license-checker/commit/28be6d4128a3625cf8ed37ba149e79ed3a41b4fa) changelog (@davglass)

### v7.1.0 (2016/09/07 15:12 +00:00)
- [9f8b7f1](https://github.com/davglass/license-checker/commit/9f8b7f1c367df68342ee5f5cc97171bf97281aa) 7.1.0 (@davglass)
- [60fcee1a](https://github.com/davglass/license-checker/commit/60fcee1a24a4ef5add98a08dd8c8a85870db3f4a6) Fixed order of operations for parsing (@davglass)
- [9d4db11](https://github.com/davglass/license-checker/commit/9d4db112794b8110a1bfe0abfe2d3e3b3812299) Updated changelog (@davglass)

### v7.0.0 (2016/09/06 14:50 +00:00)
- [a3a08da](https://github.com/davglass/license-checker/commit/a3a08da70b680d724045f655e107f36b16b45fea) 7.0.0 (@davglass)
- [b993399](https://github.com/davglass/license-checker/commit/b9933994472162cc9c4292c42f2900e17a5167f) added contrib (@davglass)
- [d783029](https://github.com/davglass/license-checker/commit/d7830297b053496e45b7145b3cef5809ac8ab1d1) Fixes #76 - If readme data not found, look for it on disk (@davglass)
- [#75](https://github.com/davglass/license-checker/pull/75) Merge pull request #75 from revov/master (@revov)
- [6204a24](https://github.com/davglass/license-checker/commit/6204a244ee71ae0329b4f2c1ca6cf37713741f32)
  Fix incorrect example in README (@revov)
- [90e24ac](https://github.com/davglass/license-checker/commit/90e24ac9ad20591d5ac166cc2974034fb997b30a)
  Added changelog (@davglass)
- [dd768c6](https://github.com/davglass/license-checker/commit/dd768c6c39b596881a2eca582ad4bb818679)
  Added contributors (@davglass)

### v6.0.0 (2016/08/26 19:46 +00:00)
- [4bca140](https://github.com/davglass/license-checker/commit/4bca140db752b7b98ad1e261e5ee965d84995741)
  6.0.0 (@davglass)
- [889ff17](https://github.com/davglass/license-checker/commit/889ff17aeb01a2180f4c9100ca9ea528a264d81)
  updating tests (@davglass)
- [ed18801](https://github.com/davglass/license-checker/commit/ed188010351c4af44d0b3d2da1025e53bf10f)
  Merge branch 'nodeback' of git://github.com/bengl/license-checker into bengl-nodeback (@davglass)
- [34bb266](https://github.com/davglass/license-checker/commit/34bb26677cd0e25500a545d4f1e8ae216b8cc7)
  dropping node 0.10 and adding node 6.0 to travis (@davglass)
- [1d30667](https://github.com/davglass/license-checker/commit/1d306674a7fface2cad31f2bd76d7584c7922bce)
  updating tests (@davglass)
- [65d7c73](https://github.com/davglass/license-checker/commit/65d7c73f967111003fcc71ae24a4a5391bd48046)
  Cleaned up some code (@davglass)
- [753998](https://github.com/davglass/license-checker/commit/f753998a06cbb5f5d9ff6430b3588a3b3c3f81339)
  Merge branch 'master' of git://github.com/amittkSharma/license-checker into amittkSharma-master (@davglass)
- [28e4462](https://github.com/davglass/license-checker/commit/28e4462543724685080c1623dc2dbd57cab1c30)
  Added tests and cleaned up linting (@davglass)
- [153857a](https://github.com/davglass/license-checker/commit/153857a251a97d14c883f647957944b2980b5aa)
  Publisher Information incorporated.
- [1265cb3](https://github.com/davglass/license-checker/commit/1265cb38d186ab1609dbdafa4124f237f8aa15846)
  Added ability to output individual license files (@aweber1)
- [49e11d0](https://github.com/davglass/license-checker/commit/49e11d049645d65b7ef6d3c778af54e13af888)
  fix tests (@bengl)
- [26ffc6f](https://github.com/davglass/license-checker/commit/26ffc6f63443c372c9a189f0c137608b3d5e4fa)
  use nodeback/errback (@bengl)

### v5.1.2 (2016/05/04 18:32 +00:00)
- [7101c77](https://github.com/davglass/license-checker/commit/7101c7767ad134fe0f02c298e7df2a4080e311)
  5.1.2 (@davglass)
- [#67](https://github.com/davglass/license-checker/pull/67) Merge pull request #67 from graingert/patch-2
  (@graingert)
- [a6ee290](https://github.com/davglass/license-checker/commit/a6ee2904ebf7cced220dcefe8548f80ec74fded3a2)
  Upgrade read-installed (@graingert)
- [42e29c6](https://github.com/davglass/license-checker/commit/42e29c637940959121842762d74921bece5e8d93)
  migrating tests from vows to mocha (@davglass)

### v5.1.1 (2016/04/14 13:20 +00:00)
- [6ae0334](https://github.com/davglass/license-checker/commit/6ae03348ca8e5835ac732028af dab664a381cfc4)
  5.1.1 (@davglass)
- [0720a46](https://github.com/davglass/license-checker/commit/0720a46569f3e12eb398165526944f307e08aad6)
  Fixes #64 - Turn off color for --json/markdown/csv options (@davglass)
### v5.1.0 (2016/03/09 15:50 +00:00)
- [86772d2](https://github.com/davglass/license-checker/commit/86772d202b63d4a4c6729b9a774193ca6ef5cb3f) 5.1.0 (@davglass)
- [6f1](https://github.com/davglass/license-checker/pull/61) Merge pull request #61 from sgtduck (https://github.com/davglass/license-checker/commit/6f1) 5.1.0 (@davglass)
- [6547ac1](https://github.com/davglass/license-checker/commit/6547ac1ded58560c7c4abdcf68fa6d9178b8f1e) Removed obsolete [key].name assignment from tests (@davglass)
- [00b18a6](https://github.com/davglass/license-checker/commit/00b18a6012387ea14a8284c07bf2d08740e03dd) Added a --relativeLicensePath option to display the license file relatively to the current path. (@sgtdck)
- [629da20](https://github.com/davglass/license-checker/commit/629da201e0d1170057fd9e1e05efcb9323f1bda) Added latest contributors (@davglass)

### v5.0.1 (2016/02/08 15:57 +00:00)
- [33731fc](https://github.com/davglass/license-checker/commit/33731fc4a829367a148ffe3fca2f3d41e3a625) 5.0.1 (@davglass)
- [fe06eae](https://github.com/davglass/license-checker/commit/fe06eae02f8844d050ad0f80d576a221dd605131de) Fixed replace (@davglass)
- [997914f](https://github.com/davglass/license-checker/commit/997914fd6f94030452d4d9bf549eb156ae8ef7c) fix git+https URLs (@graingert)
- [#57](https://github.com/davglass/license-checker/pull/57) Merge pull request #57 from Damien-Larmine/doc-license-asterisk (@Damien-Larmine)
- [2adff2d](https://github.com/davglass/license-checker/commit/2adff2da98a5af79479a162e4998eb5892ab31c3) Explicit the asterisk in license report (@Damien-Larmine)
- [b2fc9cf](https://github.com/davglass/license-checker/commit/b2fc9cf2a8ea008e7593037d046f245a7ce7f25) 5.0.0 (@davglass)
- [7f71c53](https://github.com/davglass/license-checker/commit/7f71c53833cf1318f8e84215fcb1291f9ce4fbd5) updated contributors (@davglass)

### v5.0.0 (2016/01/22 15:31 +00:00)
- [55](https://github.com/davglass/license-checker/commit/55) Merge pull request #55 from davglass/devflag (@davglass)
- [ff1d8f2](https://github.com/davglass/license-checker/commit/ff1d8f20d57248a4f61e2fc9279fdfe91a46c) supporting --production and --development flags (@davglass)
- [#53](https://github.com/davglass/license-checker/pull/53) Merge pull request #53 from swashcap/enhancement/#46-verbose-output (@swashcap)
- [54](https://github.com/davglass/license-checker/commit/54) Merge pull request #54 from davglass/travis-updates (@davglass)
- [2c80d9d](https://github.com/davglass/license-checker/commit/2c80d9d2d5c59340698630142560969a1e23f22) fixing tests in other node versions (@davglass)
- [78820ac](https://github.com/davglass/license-checker/commit/78820ac393974069a1e23f22) Travis config updates (@davglass)
- [94263b](https://github.com/davglass/license-checker/commit/94263b203046e2f53d64997b0261498010eb38d) Update documentation for debug use. (@swashcap)
- [429e44d](https://github.com/davglass/license-checker/commit/429e44d69752211ef22bb03af9df67820015446f) Change `console.error` call to use debug module. (@swashcap)
- [7312daf](https://github.com/davglass/license-checker/commit/7312daf7af195a1b9d1bf290fae15963fbb2ab4) Add debug to project. (@swashcap)
### v4.2.0 (2015/10/20 13:44 +00:00)
- [3583d92](https://github.com/davglass/license-checker/commit/3583d925e96c9f63b3bbf0110b11e8448aed0e7) 4.2.0 (@davglass)
- [981eeda](https://github.com/davglass/license-checker/commit/981eeda898084dea88d54ea29c5bda93c5313fb6) Fixes #49: Default to unknown if licenses null and added ISC support (@davglass)

### v4.1.2 (2015/09/29 13:46 +00:00)
- [ba946dd](https://github.com/davglass/license-checker/commit/ba946dd1065e762c36b27dcecd9bb7389ad18d4) 4.1.2 (@davglass)
- [#48](https://github.com/davglass/license-checker/pull/48) Merge pull request #48 from CantemoInternal/master (@CantemoInternal)
- [e314a6d](https://github.com/davglass/license-checker/commit/e314a6d1f505d19478c33871640a763fabff6abd) Fix json, csv and markdown output for license-checker (@mattiasa)

### v4.1.1 (2015/09/12 14:39 +00:00)
- [c918314](https://github.com/davglass/license-checker/commit/c918314bfc687b6471d288bb5857dd1caac1393) 4.1.1 (@davglass)
- [#47](https://github.com/davglass/license-checker/pull/47) Merge pull request #47 from tmpvar/ensure-bin-runs (@tmpvar)
- [a842eca](https://github.com/davglass/license-checker/commit/a842ecae4f008c07df3039dcaac4641d7273625) add missing ) (@tmpvar)
- [1d513bb](https://github.com/davglass/license-checker/commit/1d513bb673d726f34286061446f965e46d96170e) ensure bin/license-checker works at a high-level (@tmpvar)

### v4.1.0 (2015/09/09 14:52 +00:00)
- [a4bb437](https://github.com/davglass/license-checker/commit/a4bb437ffa806c0832f877dada2de7a708763d0a1f) 4.1.0 (@davglass)
- [ddd317a](https://github.com/davglass/license-checker/commit/ddd317a5a09344f0021812b8dee2f39bbbf07a0) fixed all of the windows line endings (@davglass)
- [11e6487](https://github.com/davglass/license-checker/commit/11e64870484435e1e6f8efb9b0f136b2f737a6b) Fix issues caused by merge (@ptusch)
- [313a7ed](https://github.com/davglass/license-checker/commit/313a7edc3dbdf8d50e43ba8da89306d3e89326bc) Merge remote-tracking branch 'remotes/upstream/master' into ft_custom_format (@ptusch)
- [1f5e326](https://github.com/davglass/license-checker/commit/1f5e326d8ca91fc8a396b9e02e3eac9df62a82) added contributors to package.json (@davglass)
- [#45](https://github.com/davglass/license-checker/pull/45) Merge pull request #45 from hufsm/ft_error_handling (@hufsm)
- [c806034](https://github.com/davglass/license-checker/commit/c806034ab6f6eacf297d0c23af536b290ba8c81) Remove unused var statement (@ptusch)
- [778b1fa](https://github.com/davglass/license-checker/commit/778b1fa1a9e1be0e816545165020222d2dbf17ef) More tests to fulfill the required % (@ptusch)
- [22257d9](https://github.com/davglass/license-checker/commit/22257d95490765db816578cc48c2b5b62639f1aa) Add tests for licenses (WTFPL) (@ptusch)
- [0fe7561](https://github.com/davglass/license-checker/commit/0fe75611a73970fe9e9e0aaab2d4bdee5250435b2) Remove unused filter option (never used) (@ptusch)
- [83ca10](https://github.com/davglass/license-checker/commit/83ca10493a5a05f9ba6e080b5c03f9c4f65d6) Change license shortcut for WTFPL (@ptusch)
- [3e348db](https://github.com/davglass/license-checker/commit/3e348dbe5feeb7bd356ebeff3cd34e69329d8c8c1)
More tests for the custom formatter (@ptusch)
- [96c2df6](https://github.com/davglass/license-checker/commit/96c2df624c509df2bf3f6bd8b83a613f8ae89d9e8c) Add tests for customFormatter (@ptusch)
- [c9948c4](https://github.com/davglass/license-checker/commit/c9948c4b7b2ccced053ba63f4fd2a6a6d0a46493) Add test configs (@ptusch)
- [4b2dea0](https://github.com/davglass/license-checker/commit/4b2dea0c12f6faba63adb2129625825c3b5a6570) Fix merge issues - semicolon slipped through (@ptusch)
- [cc7662a](https://github.com/davglass/license-checker/commit/cc7662a46719c97745410e3ac2b782f937dc54ef) Merge branch 'master' into ft_error_handling (@ptusch)
- [04a2294](https://github.com/davglass/license-checker/commit/04a2294b82a8227cccb83d76f73cb7cf62af533) Add additional test (@ptusch)
- [52fa91c](https://github.com/davglass/license-checker/commit/52fa91c5cb3c45736e6e9e7f400a27a773502d2) Add parseJson test (@ptusch)
- [d846c5e](https://github.com/davglass/license-checker/commit/d846c5e6970737dd3085688e7b601e9ea4546417) Fix tests after merge conflict (@ptusch)
- [7ead458](https://github.com/davglass/license-checker/commit/7ead4583f803b5bda4e9f66768bbb457fe8de44) Merge branch 'master' into ft_custom_format (@ptusch)
- [4aa7065](https://github.com/davglass/license-checker/commit/4aa7065c5eef88d49c3684369b2edd9802c3f374) Add tests for customFormat (@ptusch)

### v4.0.0 (2015/09/04 13:40 +00:00)
- [3fbba5b](https://github.com/davglass/license-checker/commit/3fbba5bd5a489ea8e17328eabf71c4d5e9beb17e) 4.0.0 (@davglass)
- [ce8d661](https://github.com/davglass/license-checker/commit/ce8d66108a49af7855c16231ce6675c567a2ffe7f) tests: added 100% coverage for license file (@davglass)
- [#42](https://github.com/davglass/license-checker/pull/42) Merge pull request #42 from ptusch/ft_onlyunknown (@ptusch)
- [8929e90](https://github.com/davglass/license-checker/commit/8929e90413be0362f19e755048e9218c372abf68) Add tests for the error generation (@ptusch)
- [1a93a41](https://github.com/davglass/license-checker/commit/1a93a41569a53a1580c942c903a5bc127ad2692) Add error handling (@ptusch)
- [6b79145](https://github.com/davglass/license-checker/commit/6b791457463c71b22b09e0ecff16bf9f82db682) Add additional check to negate the test (@ptusch)
- [559e72c](https://github.com/davglass/license-checker/commit/559e72cb2420bc42bf28b498702dad13e9f67a) Add test for the first function call. (@ptusch)
- [fa8fffd](https://github.com/davglass/license-checker/commit/fa8fffd69fd8ae475484a9b757a7b6ea738856) Remove trivial UNKNOWN check (@ptusch)
- [6fa44ef](https://github.com/davglass/license-checker/commit/6fa44ef8c045ab2b0a1291b45990edf52adfc0e2) Merge branch 'master' into ft_custom_format (@ptusch)
- [efe23d9](https://github.com/davglass/license-checker/commit/efe23d991cfcb40dc6980bfb13461408da822b1f) Merge branch 'master' into ft_onlyunknown (@ptusch)
- [c287959](https://github.com/davglass/license-checker/commit/c2879592858805d7ecd0304a39c5cf079e06a7c) Add option to only show unknown licenses (@ptusch)
- [#43](https://github.com/davglass/license-checker/pull/43) Merge pull request #43 from senotrusov/master (@senotrusov)
- [d7a663f](https://github.com/davglass/license-checker/commit/d7a663f9deeb4d7a648a6d2c9db84d5667ee540) Add test for exclude option. (@senotrusov)
- [45a27de](https://github.com/davglass/license-checker/commit/45a27de02a3c1d57a86e3f8e14b6e335788c23ba)
Add option to exclude from the output modules which licenses are in the comma-separated list (@senotrusov)
- [#41](https://github.com/davglass/license-checker/pull/41) Merge pull request #41 from ptusch/ft_path_to_unknown_dependency (@ptusch)
- [1ff5f22](https://github.com/davglass/license-checker/commit/1ff5f225fd2ba97edf3ef3f6462981832740c19) Add option to see the path of a dependency if unknown (@ptusch)
- [ea4d629](https://github.com/davglass/license-checker/commit/ea4d6297de32d94201bb4786da7ba824d928c8f2) Add additional 'undefined' checks (@ptusch)
- [af043a2](https://github.com/davglass/license-checker/commit/af043a2c11ae4d528bfe71914cd10672855b3297) Merge branch 'master' into ft_custom_format (@ptusch)
- [2d94a19](https://github.com/davglass/license-checker/commit/2d94a19e3e77836345aa227893a82d956141b2e) updating the docs for --unknown to make it clear (@davglass)
- [38a2bd2](https://github.com/davglass/license-checker/commit/38a2bd28a9597f7db06d844258f1070054b388b4c) Fix codestyle and whitespaces (@ptusch)
- [dd173ba](https://github.com/davglass/license-checker/commit/dd173badc65368de6140476ffe5ae3768dfb9618) Add --customPath in README.md (@ptusch)
- [911fc1c](https://github.com/davglass/license-checker/commit/911fc1e6b5ce27a3a2a6c60b04b490c08be138bc070) Finalize the customFormat (markdown) (@ptusch)
- [8102091](https://github.com/davglass/license-checker/commit/8102091f7949fa8b73c8f03a4ead3624b81148) Add functionality to add a custom format (@ptusch)

### v3.1.0 (2015/06/29 14:25 +00:00)
- [59e1e58](https://github.com/davglass/license-checker/commit/59e1e58ca0bea1cd38ed3a1712fe7ca624d860b0) 3.1.0 (@davglass)
- [0e33f7a](https://github.com/davglass/license-checker/commit/0e33f7a2cb9fae69d13d0e2b5c69f70885e09e08) reformatted code, fixed lint issues and added tests (@davglass)
- [83a7f09](https://github.com/davglass/license-checker/commit/83a7f0933d1ccf331279e53fccee13e4efface91) Merge branch 'asMarkdown' of git://github.com/ptusch/license-checker into ptusch-asMarkdown (@davglass)
- [703ad47](https://github.com/davglass/license-checker/commit/703ad4714173559e05267e8fd24c74eb9d3aee0) fixing tests (@davglass)
- [2d9fffd](https://github.com/davglass/license-checker/commit/2d9fffd75192e9812692c9099481d23e265bb493) Added Markdown Option (@ptusch)
- [67dca36](https://github.com/davglass/license-checker/commit/67dca365b941a7265e4a98d2de533b22c761b169) Remove color tags (@ptusch)

### v3.0.3 (2015/06/03 13:35 +00:00)
- [7475c12](https://github.com/davglass/license-checker/commit/7475c129bf291289e4616d962a537401e8) 3.0.3 (@davglass)
- [c7dae0f](https://github.com/davglass/license-checker/commit/c7dae0fe5317a17b3ae6f61e8c118fabc6bb43a56) spdx license (@davglass)

### v3.0.2 (2015/05/12 20:29 +00:00)
- [01f3b53](https://github.com/davglass/license-checker/commit/01f3b5384998be6243d38e0d95a4478e5fe3cfa5) 3.0.2 (@davglass)
- [fc0083](https://github.com/davglass/license-checker/commit/fc00832ee291c55e53efe6b37a7b9d89f191bc7) defensive code, fixes #34 (@davglass)

### v3.0.1 (2015/05/04 12:28 +00:00)
- [0479570](https://github.com/davglass/license-checker/commit/0479570fa87eccc9b37ffbb25aafaef7aaec6636)
3.0.1 (@davglass)
- [d85e355](https://github.com/davglass/license-checker/commit/d85e355b55998952facf202eabf8d374f6ec69a7e) Checking for valid path first (@davglass)

#### v3.0.0 (2015/04/24 14:38 +00:00)
- [e22be61d](https://github.com/davglass/license-checker/commit/e22be61d532fe7ff1b8627f55a97e1863a78096) 3.0.0 (@davglass)
- [fde9635](https://github.com/davglass/license-checker/commit/fde96353dd207eca4badffbe3e0d033a038f17b5) always try to find and log the license file (@davglass)

#### v2.0.2 (2015/01/26 18:18 +00:00)
- [490f31d](https://github.com/davglass/license-checker/commit/490f31d6820f068814ce27aaf3a1ac6d58f733) 2.0.2 (@davglass)
- [a3fefdb](https://github.com/davglass/license-checker/commit/a3fefdb8afaf7d64714ddefb793d757a9e8ad065) fixes #30 (@davglass)

#### v2.0.1 (2015/01/12 18:34 +00:00)
- [898fbd5](https://github.com/davglass/license-checker/commit/898fbd53fa289338cf8a9a1735eea440c2a6c525) 2.0.1 (@davglass)
- [d47dbe4](https://github.com/davglass/license-checker/commit/d47dbe45c992dbdb15bfb45e8af7f18ee0ca737) fixes #29 (@davglass)

#### v2.0.0 (2014/12/04 16:34 +00:00)
- [f4c7b1a](https://github.com/davglass/license-checker/commit/f4c7b1a3d4e9e4753a9b267e51714fe7f051e) syntax cleanup (@davglass)
- [d189623](https://github.com/davglass/license-checker/commit/d1896234a5649621ee57ff9a901877d76921794) Document CLI options in Readme. (@timoxley)
- [11e204c](https://github.com/davglass/license-checker/commit/11e204c5c078eae1f65fa010达229d2a9fb4b5) Default to --no-colour for json, csv & file output. (@timoxley)
- [7e80d50](https://github.com/davglass/license-checker/commit/7e80d50c733f8ee42ce7b4a79dc3344b3dd330cffffff) Add --out flag for writing an output file. (@timoxley)
- [47bff9c](https://github.com/davglass/license-checker/commit/47bff9c9e0e249b2847c8ed4f3eaadb0dcdd731b9) Always print to stdout, don't write files. (@timoxley)
- [b725080](https://github.com/davglass/license-checker/commit/b725080d2dbdfdb0f994867577d4c5e3de0ee2a) Don't print info messages to stdout. (@timoxley)

#### v1.0.1 (2014/11/27 15:23 +00:00)
- [e60f792](https://github.com/davglass/license-checker/commit/e60f792159351a7ccf1f7af73e289af6e26a5) 1.0.1 (@davglass)
- [3f74178](https://github.com/davglass/license-checker/commit/3f7417822de88049f8d1baf4ed84190d327df15c) Merge branch 'fixinfinite recursion' of github.com:yzapuchlak/license-checker into yzapuchlak-fixinfinite recursion (@davglass)
- [d78ea0](https://github.com/davglass/license-checker/commit/d78ea0dca546283440c89fec2dc6818dc4c1c1) sticking vows 0.8.0 for testing (@davglass)
Fixed issue where circular dependencies in dependency tree would cause the flatten function to infinitely recurse. If the current key is found to be already in the data object (which means it was already processed), we stop processing that key and return the data object. (@yzapuchlak)

### v1.0.0 (2014/11/21 13:56 +00:00)
- [917193d](https://github.com/davglass/license-checker/commit/917193dab27147534976b71307a77f48198ca322) 1.0.0 (@davglass)
- [63e5f9c](https://github.com/davglass/license-checker/commit/63e5f9c77e348720b3867171bce67541e846d088) added options.filter to checker.init (@davglass)

### v0.1.2 (2014/11/20 15:48 +00:00)
- [8e5f5f0](https://github.com/davglass/license-checker/commit/8e5f5f0e7f83252e4039f6c7d16ce266acc1a33a) 0.1.2 (@davglass)
- [3d7a74e](https://github.com/davglass/license-checker/commit/3d7a74e90122f908057b2008ab29d8bd49011291) fixed lint issue (@davglass)

### v0.1.1 (2014/11/20 15:47 +00:00)
- [750d507](https://github.com/davglass/license-checker/commit/750d5079d32b44ec984ff7ed672498d1b197235) 0.1.1 (@davglass)
- [94522e0](https://github.com/davglass/license-checker/commit/94522e099434dfdf163c3f2c5cacb3e6aa196cf8) fixed merge issues (@davglass)

### v0.1.0 (2014/11/20 15:41 +00:00)
- [8127f5e](https://github.com/davglass/license-checker/commit/8127f5e79eececb7df86cd5f694080e4183dc81) 0.1.0 (@davglass)
- [76c71ac](https://github.com/davglass/license-checker/commit/76c71ac46baeca8c349fe5892cb659e859b9608) Updated to newer read-installed (@davglass)

### v0.0.10 (2014/08/12 12:48 +00:00)
- [807507d](https://github.com/davglass/license-checker/commit/807507df1a293b0c360adff7d25af911331ff2b) 0.0.10 (@davglass)
- [e39ad9b](https://github.com/davglass/license-checker/commit/e39ad9bce6bcd545772c241e2e7aba8f40d5bbbb) Added checking that the license file is a normal file, because otherwise an appropriately named directory crashes the license-checker through EISDIR, illegal operation on a directory.
- [40ca4da](https://github.com/davglass/license-checker/commit/40ca4dad0c163ca774c87b1de249b2d95952faa9) colorize (@couchand)

### v0.0.9 (2014/07/29 01:48 +00:00)
- [c14ef79](https://github.com/davglass/license-checker/commit/c14ef79f10f950f0985e7e0b3138147b98ab1ec) 0.0.9 (@davglass)
- [ba1b66b](https://github.com/davglass/license-checker/commit/ba1b66b172a2f9f8dbcde56f40206820bc0916b) Fixes #12 - Using some nicer regexes (@davglass)
- [bce94d5](https://github.com/davglass/license-checker/commit/bce94d5a8ab2f146356851776b2543df65fbf750) case insensitive check (@davglass)
- [87c3406](https://github.com/davglass/license-checker/commit/87c3406aee2507eaf1d00d5926dcb59a16e7c48) Allow for LICENCE (@gdw2)
### v0.0.8 (2014/07/07 20:10 +00:00)
- [6ad6ee2](https://github.com/davglass/license-checker/commit/6ad6ee21813e698e7f995e98dcadafbd961f6a33) 0.0.8 (@davglass)
- [b8c5815](https://github.com/davglass/license-checker/commit/b8c58151135f579f1bf0d1cfcf194d034e012073f) added tests and refactored the code a little (@davglass)
- [275f276](https://github.com/davglass/license-checker/commit/275f276dc6e6d635f9c5b7e7a978e286f2d174) Added --csv option to readme file
- [d91eb72](https://github.com/davglass/license-checker/commit/d91eb72784966853ae6e8844b953b4d3d9ebe5) Apparently abbrev is now at version 1.0.5 which caused the unit test to fail.
- [40b4a42](https://github.com/davglass/license-checker/commit/40b4a420c520c2008b9395d1a8fd38b56f846f5) Modified changes based on lshint output during Travis build
- [54e2390](https://github.com/davglass/license-checker/commit/54e2390cefd1dc664a15cd3a84347227889f48b) Added csv output option similar to the existing JSON output option. Requires an output filename.

### v0.0.7 (2014/02/26 04:52 +00:00)
- [1ce6d42](https://github.com/davglass/license-checker/commit/1ce6d42147b2d0237807818c8059003b331816cf) 0.0.7 (@davglass)
- [7d04195](https://github.com/davglass/license-checker/commit/7d041959450c68b3e5f427007c436c67c3ad6b) Added not check and a proper test (@davglass)
- [40d6e25](https://github.com/davglass/license-checker/commit/40d6e2580496c33ee1d408eb28ff523852e8eb2) Undefined handling (@garrows)
- [90217d0](https://github.com/davglass/license-checker/commit/90217dd070d626c76eb47b1610ed5ee6ae840d) travis Stuff (@davglass)
- [2bd76e8](https://github.com/davglass/license-checker/commit/2bd76ec94f740cd83f62460e07341382c6a32d8b) 0.0.6 (@davglass)
- [6f390ff](https://github.com/davglass/license-checker/commit/6f390ff66b31a07939e6325796bdcf6eb5907) Added some better tests (@davglass)
- [2dbcac3](https://github.com/davglass/license-checker/commit/2dbcac37b5795ceb6db53452d0e41d1c4bc3153) Looking for WTFPL as well as the string (@davglass)
- [5d91a07](https://github.com/davglass/license-checker/commit/5d91a073a8b766c9a497c1e3b83198ce97e144f) Flattened array of one (@davglass)
- [e95da82](https://github.com/davglass/license-checker/commit/e95da820c63e0d56c4567f1cfadf07142d3110eca) Fixed extra var (@davglass)
- [fe8f2f7](https://github.com/davglass/license-checker/commit/fe8f2f7d39d520c04d873d349d4f5559544222) Removing unnecessary log statement (@paulmand31)
- [72a6f24](https://github.com/davglass/license-checker/commit/72a6f24034e3ecf5ec07bb504cb931c0e536fa3) Adding license checks in the LICENSE file (and variations) (@paulmand31)
- [#7](https://github.com/davglass/license-checker/pull/7) Merge pull request #7 from jamesbloomer/master (@jamesbloomer)
- [a81ee0](https://github.com/davglass/license-checker/commit/a81ee0a27f580688e371cc15015e848591bb92) Fixed missing "install" in npm -g (@jamesbloomer)
- [416b012](https://github.com/davglass/license-checker/commit/416b01219e0a8274e05cd833e05037902adeb) Updated Version (@davglass)
- [6d9bd1e](https://github.com/davglass/license-checker/commit/6d9bd1e195202c871693d4b1e9bac8c055683e8b) Fixed lint issue and added a test placeholder. Will add more tests soon (@davglass)
- [655a316](https://github.com/davglass/license-checker/commit/655a316f6448df3a449f88b88e8878a95a8b153e) Added Travis test file (@davglass)
- [99895df](https://github.com/davglass/license-checker/commit/99895df6d3ac0f9ae4e8e935311fb1b3f88a688e)
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

// Original can be found at:
// https://bitbucket.org/lindenlab/lsl
// Modifications by Joshua Bell inexorabletash@gmail.com
// https://github.com/inexorabletash/polyfill

// ES3/ES5 implementation of the Khronos Typed Array Specification
// Ref: http://www.khronos.org/registry/typedarray/specs/latest/
// Date: 2011-02-01
//
// Variations:
// * Allows typed_array.get/set() as alias for subscripts (typed_array[])

The MIT License (MIT)

Copyright (c) 2017 CoderPuppy

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
CC0 1.0 Universal

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;

ii. moral rights retained by the original author(s) and/or performer(s);

iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;

iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;

v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and

vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

a. No trademark or patent rights held by Affirmer are waived, abandoned,
surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

For more information, please see <http://creativecommons.org/publicdomain/zero/1.0/>

Copyright 2014 jQuery Foundation and other contributors
http://jquery.com/

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012-2014 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter
Copyright (c) 2014-2015 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

source: https://github.com/johan/world.geo.json
license:
Unless otherwise noted in individual files or directories, this is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>
The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided
that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

b) use the modified Package only within your corporation or organization.

c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

b) accompany the distribution with the machine-readable source of the Package with your modifications.

c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall
under the copyright of this Package, but belong to whoever generated
them, and may be sold commercially, and may be aggregated with this
Package. If such scripts or library files are aggregated with this
Package via the so-called "undump" or "unexec" methods of producing a
binary executable image, then distribution of such an image shall
neither be construed as a distribution of this Package nor shall it
fall under the restrictions of Paragraphs 3 and 4, provided that you do
not represent such an executable image as a Standard Version of this
Package.

7. C subroutines (or comparably compiled subroutines in other
languages) supplied by you and linked into this Package in order to
emulate subroutines and variables of the language defined by this
Package shall not be considered part of this Package, but are the
equivalent of input as in Paragraph 6, provided these subroutines do
not change the language in any way that would cause it to fail the
regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always
permitted provided that the use of this Package is embedded; that is,
when no overt attempt is made to make this Package's interfaces visible
to the end user of the commercial distribution. Such use shall not be
construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote
products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End
Copyright 2009 Nokia Corporation

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next
paragraph) shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Google LLC and its affiliates (“Google”) own all legal right, title and interest in and to the content decryption module software (“Software”) and related documentation, including any intellectual property rights in the Software. You may not use, modify, sell, or otherwise distribute the Software without a separate license agreement with Google. The Software is not open source software.

If you are interested in licensing the Software, please contact www.widevine.com

### v4.0.1 (2016-12-14)

#### WHOOPS

* [fb9b1ce](https://github.com/npm/nopt/commit/fb9b1ce57b3c69b4f7819015be87719204f77ef6)
  Merged so many patches at once that the code fencing ([@adius](https://github.com/adius)) added got broken. Sorry, ([@adius](https://github.com/adius))!
  ([@othiym23](https://github.com/othiym23))

### v4.0.0 (2016-12-13)

#### BREAKING CHANGES

* [651d447](https://github.com/npm/nopt/commit/651d4473946096d341a480bbe56793de3fc706aa)
  When parsing String-typed arguments, if the next value is """, don't simply swallow it. ([@samjonester](https://github.com/samjonester))

#### PERFORMANCE TWEAKS

* [3370ce8](https://github.com/npm/nopt/commit/3370ce87a7618ba228883861db84dbcdf252a9)
  Simplify initialization. ([@elidoran](https://github.com/elidoran))
* [356e58e](https://github.com/npm/nopt/commit/356e58e3b3b431a4b1af7fd7bdee44c2c0526a09)
  Store `Array.isArray(types[arg])` for reuse. ([@elidoran](https://github.com/elidoran))
* [0d95e90](https://github.com/npm/nopt/commit/0d95e90515844f266015b56d2e80b94e5d14a07e)
  Interpret single-item type arrays as a single type. ([@samjonester](https://github.com/samjonester))
* [07c69d3](https://github.com/npm/nopt/commit/07c69d38b5186450941fbb505550becb78a0e925)
  Simplify key-value extraction. ([@elidoran](https://github.com/elidoran))
* [39b6e5c](https://github.com/npm/nopt/commit/39b6e5c65ac47f60cd43a1fbeece5cd4c834c254)
  Only call `Date.parse(val)` once. ([@elidoran](https://github.com/elidoran))
* [934943d](https://github.com/npm/nopt/commit/934943dfecb55123a2b15959fe2a359319a5dbd)
  Use `os.env.home()` to find a user's home directory instead of assuming it's always `$HOME`. ([@othiym23](https://github.com/othiym23))
### TEST & CI IMPROVEMENTS

* ['326ffff'](https://github.com/npm/nopt/commit/326ffff7f78a00bcd316adecf69075f8a8093619)
  Fix `\tmp` test to work on Windows.
  (@elidoran)

* ['c89d31a'](https://github.com/npm/nopt/commit/c89d31a49d14f2238bc6672db08da697b57f1b)
  Only run Windows tests on Windows, only run Unix tests on a Unix.
  (@elidoran)

* ['affd3d1'](https://github.com/npm/nopt/commit/affd3d1d0addffa93006397b2013b18447339366)
  Refresh Travis to run the tests against the currently-supported batch of npm
  versions. (@helio)

* ['55f9449'](https://github.com/npm/nopt/commit/55f94497d163ed4d16dd55fd6c4fb95cc440e66d)
  `\tap@8.0.1` (@othiym23)

### DOC TWEAKS

* ['5271229'](https://github.com/npm/nopt/commit/5271229ee7c810217dd51616c086f5d9ab224581)
  Use JavaScript code block for syntax highlighting.
  (@adius)

* ['c0d136f'](https://github.com/npm/nopt/commit/c0d136f229f9994c5dfcec4a888edeff7a07682)
  The code sample in the README had `\many2: \[ oneThing \]`, and now it has
  `\many2: \[ two, things \]`. (@silkentrance)

---

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2010 Julian R Seward. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@bzip.org
bzip2/libbzip2 version 1.0.6 of 6 September 2010

________________________________________________________
ISC License

Copyright (c) 2018, Andrea Giammarchi, @WebReflection

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License

Copyright (C) 2019 Erik Barke, Monounity.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2010-2015 Benjamin Peterson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2017 The Polymer Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 1997 Gregory Pietsch

[These files] are hereby placed in the public domain without restrictions. Just
give the author credit, don't claim you wrote it or prevent anyone else from
using it.
LZMA SDK is placed in the public domain.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE

===================================================================

PNG Reference Library License version 2

===================================================================

* Copyright (c) 1995-2019 The PNG Reference Library Authors.
* Copyright (c) 2018-2019 Cosmin Truta.
* Copyright (c) 1996-1997 Andreas Dilger.
* Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind,
express or implied, including, without limitation, the warranties
of merchantability, fitness for a particular purpose, title, and
non-infringement. In no event shall the Copyright owners, or
anyone distributing the software, be liable for any damages or
other liability, whether in contract, tort or otherwise, arising
from, out of, or in connection with the software, or the use or
other dealings in the software, even if advised of the possibility
of such damage.

Permission is hereby granted to use, copy, modify, and distribute
this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented: you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabuddhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are
Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

   Tom Lane
   Glenn Randers-Pehrson
   Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

   John Bowler
   Kevin Bracey
   Sam Bushell
   Magnus Holmgren
   Greg Roelofs
   Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

   Andreas Dilger
   Dave Martindale
   Guy Eric Schalnat
   Paul Schmidt
   Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions thereof, for any purpose, without fee, subject
to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

APPLE PUBLIC SOURCE LICENSE Version 2.0 - August 6, 2003

Please read this License carefully before downloading this software. By downloading or using this software, you are agreeing to be bound by the terms of this License. If you do not or cannot agree to the terms of this License, please do not download or use the software.

Apple Note: In January 2007, Apple changed its corporate name from "Apple Computer, Inc." to "Apple Inc." This change has been reflected below and copyright years updated, but no other changes have been made to the APSL 2.0.

1. General; Definitions. This License applies to any program or other work which Apple Inc. ("Apple") makes publicly available and which contains a notice placed by Apple identifying such program or work as "Original Code" and stating that it is subject to the terms of this Apple Public Source License version 2.0 ("License"). As used in this License:

1.1 "Applicable Patent Rights" mean: (a) in the case where Apple is the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to Apple and (ii) that cover subject matter contained in the Original Code, but only to the extent necessary to use, reproduce and/or distribute the Original Code without infringement; and (b) in the case where You are the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to You and (ii) that cover subject matter in Your Modifications, taken alone or in combination with Original Code.

1.2 "Contributor" means any person or entity that creates or contributes to the creation of Modifications.

1.3 "Covered Code" means the Original Code, Modifications, the combination of Original Code and any Modifications, and/or any respective portions thereof.

1.4 "Externally Deploy" means: (a) to sublicense, distribute or otherwise make
Covered Code available, directly or indirectly, to anyone other than You; and/or (b) to use Covered Code, alone or as part of a Larger Work, in any way to provide a service, including but not limited to delivery of content, through electronic communication with a client other than You.

1.5 "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.6 "Modifications" mean any addition to, deletion from, and/or change to, the substance and/or structure of the Original Code, any previous Modifications, the combination of Original Code and any previous Modifications, and/or any respective portions thereof. When code is released as a series of files, a Modification is: (a) any addition to or deletion from the contents of a file containing Covered Code; and/or (b) any new file or other representation of computer program statements that contains any part of Covered Code.

1.7 "Original Code" means (a) the Source Code of a program or other work as originally made available by Apple under this License, including the Source Code of any updates or upgrades to such programs or works made available by Apple under this License, and that has been expressly identified by Apple as such in the header file(s) of such work; and (b) the object code compiled from such Source Code and originally made available by Apple under this License.

1.8 "Source Code" means the human readable form of a program or other work that is suitable for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an executable (object code).

1.9 "You" or "Your" means an individual or a legal entity exercising rights under this License. For legal entities, "You" or "Your" includes any entity which controls, is controlled by, or is under common control with, You, where "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of fifty percent (50%) or more of the outstanding shares or beneficial ownership of such entity.

2. Permitted Uses; Conditions & Restrictions. Subject to the terms and conditions of this License, Apple hereby grants You, effective on the date You accept this License and download the Original Code, a world-wide, royalty-free, non-exclusive license, to the extent of Apple's Applicable Patent Rights and copyrights covering the Original Code, to do the following:

2.1 Unmodified Code. You may use, reproduce, display, perform, internally distribute within Your organization, and Externally Deploy verbatim, unmodified copies of the Original Code, for commercial or non-commercial purposes, provided that in each instance:

(a) You must retain and reproduce in all copies of Original Code the copyright
and other proprietary notices and disclaimers of Apple as they appear in the
Original Code, and keep intact all notices in the Original Code that refer to
this License; and

(b) You must include a copy of this License with every copy of Source Code of
Covered Code and documentation You distribute or Externally Deploy, and You may
not offer or impose any terms on such Source Code that alter or restrict this
License or the recipients' rights hereunder, except as permitted under Section
6.

2.2 Modified Code. You may modify Covered Code and use, reproduce, display,
perform, internally distribute within Your organization, and Externally Deploy
Your Modifications and Covered Code, for commercial or non-commercial purposes,
provided that in each instance You also meet all of these conditions:

(a) You must satisfy all the conditions of Section 2.1 with respect to the
Source Code of the Covered Code;

(b) You must duplicate, to the extent it does not already exist, the notice in
Exhibit A in each file of the Source Code of all Your Modifications, and cause
the modified files to carry prominent notices stating that You changed the files
and the date of any change; and

(c) If You Externally Deploy Your Modifications, You must make Source Code of
all Your Externally Deployed Modifications either available to those to whom You
have Externally Deployed Your Modifications, or publicly available. Source Code
of Your Externally Deployed Modifications must be released under the terms set
forth in this License, including the license grants set forth in Section 3 below, for as long as you Externally Deploy the Covered Code or twelve (12)
months from the date of initial External Deployment, whichever is longer. You
should preferably distribute the Source Code of Your Externally Deployed
Modifications electronically (e.g. download from a web site).

2.3 Distribution of Executable Versions. In addition, if You Externally Deploy
Covered Code (Original Code and/or Modifications) in object code, executable
form only, You must include a prominent notice, in the code itself as well as in
related documentation, stating that Source Code of the Covered Code is available
under the terms of this License with information on how and where to obtain such
Source Code.

2.4 Third Party Rights. You expressly acknowledge and agree that although
Apple and each Contributor grants the licenses to their respective portions of
the Covered Code set forth herein, no assurances are provided by Apple or any
Contributor that the Covered Code does not infringe the patent or other
intellectual property rights of any other entity. Apple and each Contributor
disclaim any liability to You for claims brought by any other entity based on
infringement of intellectual property rights or otherwise. As a condition to
exercising the rights and licenses granted hereunder, You hereby assume sole
responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow You to distribute the Covered Code, it is Your responsibility to acquire that license before distributing the Covered Code.

3. Your Grants. In consideration of, and as a condition to, the licenses granted to You under this License, You hereby grant to any person or entity receiving or distributing Covered Code under this License a non-exclusive, royalty-free, perpetual, irrevocable license, under Your Applicable Patent Rights and other intellectual property rights (other than patent) owned or controlled by You, to use, reproduce, display, perform, modify, sublicense, distribute and Externally Deploy Your Modifications of the same scope and extent as Apple's licenses under Sections 2.1 and 2.2 above.

4. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In each such instance, You must make sure the requirements of this License are fulfilled for the Covered Code or any portion thereof.

5. Limitations on Patent License. Except as expressly stated in Section 2, no other patent rights, express or implied, are granted by Apple herein. Modifications and/or Larger Works may require additional patent licenses from Apple which Apple may grant in its sole discretion.

6. Additional Terms. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations and/or other rights consistent with the scope of the license granted herein ("Additional Terms") to one or more recipients of Covered Code. However, You may do so only on Your own behalf and as Your sole responsibility, and not on behalf of Apple or any Contributor. You must obtain the recipient's agreement that any such Additional Terms are offered by You alone, and You hereby agree to indemnify, defend and hold Apple and every Contributor harmless for any liability incurred by or claims asserted against Apple or such Contributor by reason of any such Additional Terms.

7. Versions of the License. Apple may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Once Original Code has been published under a particular version of this License, You may continue to use it under the terms of that version. You may also choose to use such Original Code under the terms of any subsequent version of this License published by Apple. No one other than Apple has the right to modify the terms applicable to Covered Code created under this License.

8. NO WARRANTY OR SUPPORT. The Covered Code may contain in whole or in part pre-release, untested, or not fully tested works. The Covered Code may contain errors that could cause failures or loss of data, and may be incomplete or
contain inaccuracies. You expressly acknowledge and agree that use of the
Covered Code, or any portion thereof, is at Your sole and entire risk. THE
COVERED CODE IS PROVIDED "AS IS" AND WITHOUT WARRANTY, UPGRADES OR SUPPORT OF
ANY KIND AND APPLE AND APPLE’S LICENSOR(S) (COLLECTIVELY REFERRED TO AS “APPLE”
FOR THE PURPOSES OF SECTIONS 8 AND 9) AND ALL CONTRIBUTORS EXPRESSLY DISCLAIM
ALL WARRANTIES AND/OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED
to, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY, OF SATISFACTORY
QUALITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY, OF QUIET ENJOYMENT,
AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. APPLE AND EACH CONTRIBUTOR DOES NOT
WARRANT AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE COVERED CODE, THAT THE
FUNCTIONS CONTAINED IN THE COVERED CODE WILL MEET YOUR REQUIREMENTS, THAT THE
OPERATION OF THE COVERED CODE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT
DEFECTS IN THE COVERED CODE WILL BE CORRECTED. NO ORAL OR WRITTEN INFORMATION
OR ADVICE GIVEN BY APPLE, AN APPLE AUTHORIZED REPRESENTATIVE OR ANY CONTRIBUTOR
SHALL CREATE A WARRANTY. You acknowledge that the Covered Code is not intended
for use in the operation of nuclear facilities, aircraft navigation,
communication systems, or air traffic control machines in which case the failure
of the Covered Code could lead to death, personal injury, or severe physical or
environmental damage.

9. LIMITATION OF LIABILITY. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT
SHALL APPLE OR ANY CONTRIBUTOR BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT
OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE OR YOUR USE
OR INABILITY TO USE THE COVERED CODE, OR ANY PORTION THEREOF, WHETHER UNDER A
THEORY OF CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCTS LIABILITY OR
OTHERWISE, EVEN IF APPLE OR SUCH CONTRIBUTOR HAS BEEN ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF ESSENTIAL PURPOSE OF ANY
REMEDY. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF LIABILITY OF
INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS LIMITATION MAY NOT APPLY TO YOU. In
no event shall Apple's total liability to You for all damages (other than as may
be required by applicable law) under this License exceed the amount of fifty
dollars ($50.00).

10. Trademarks. This License does not grant any rights to use the trademarks
or trade names "Apple", "Mac", "Mac OS", "QuickTime", "QuickTime Streaming
Server" or any other trademarks, service marks, logos or trade names belonging
to Apple (collectively "Apple Marks") or to any trademark, service mark, logo or
trade name belonging to any Contributor. You agree not to use any Apple Marks
in or as part of the name of products derived from the Original Code or to
endorse or promote products derived from the Original Code other than as
expressly permitted by and in strict compliance at all times with Apple's third
party trademark usage guidelines which are posted at

11. Ownership. Subject to the licenses granted under this License, each
Contributor retains all rights, title and interest in and to any Modifications
made by such Contributor. Apple retains all rights, title and interest in and
to the Original Code and any Modifications made by or on behalf of Apple ("Apple
12. Termination.

12.1 Termination. This License and the rights granted hereunder will terminate:

(a) automatically without notice from Apple if You fail to comply with any term(s) of this License and fail to cure such breach within 30 days of becoming aware of such breach; (b) immediately in the event of the circumstances described in Section 13.5(b); or (c) automatically without notice from Apple if You, at any time during the term of this License, commence an action for patent infringement against Apple; provided that Apple did not first commence an action for patent infringement against You in that instance.

12.2 Effect of Termination. Upon termination, You agree to immediately stop any further use, reproduction, modification, sublicensing and distribution of the Covered Code. All sublicenses to the Covered Code which have been properly granted prior to termination shall survive any termination of this License. Provisions which, by their nature, should remain in effect beyond the termination of this License shall survive, including but not limited to Sections 3, 5, 8, 9, 10, 11, 12.2 and 13. No party will be liable to any other for compensation, indemnity or damages of any sort solely as a result of terminating this License in accordance with its terms, and termination of this License will be without prejudice to any other right or remedy of any party.


13.1 Government End Users. The Covered Code is a "commercial item" as defined in FAR 2.101. Government software and technical data rights in the Covered Code include only those rights customarily provided to the public as defined in this License. This customary commercial license in technical data and software is provided in accordance with FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for Department of Defense purchases, DFAR 252.227-7015 (Technical Data -- Commercial Items) and 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation). Accordingly, all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

13.2 Relationship of Parties. This License will not be construed as creating an agency, partnership, joint venture or any other form of legal association between or among You, Apple or any Contributor, and You will not represent to the contrary, whether expressly, by implication, appearance or otherwise.

13.3 Independent Development. Nothing in this License will impair Apple’s right to acquire, license, develop, have others develop for it, market and/or
distribute technology or products that perform the same or similar functions as, or otherwise compete with, Modifications, Larger Works, technology or products that You may develop, produce, market or distribute.

13.4 Waiver; Construction. Failure by Apple or any Contributor to enforce any provision of this License will not be deemed a waiver of future enforcement of that or any other provision. Any law or regulation which provides that the language of a contract shall be construed against the drafter will not apply to this License.

13.5 Severability. (a) If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of the License will be enforced to the maximum extent permissible so as to effect the economic benefits and intent of the parties, and the remainder of this License will continue in full force and effect. (b) Notwithstanding the foregoing, if applicable law prohibits or restricts You from fully and/or specifically complying with Sections 2 and/or 3 or prevents the enforceability of either of those Sections, this License will immediately terminate and You must immediately discontinue any use of the Covered Code and destroy all copies of it that are in your possession or control.

13.6 Dispute Resolution. Any litigation or other dispute resolution between You and Apple relating to this License shall take place in the Northern District of California, and You and Apple hereby consent to the personal jurisdiction of, and venue in, the state and federal courts within that District with respect to this License. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded.

13.7 Entire Agreement; Governing Law. This License constitutes the entire agreement between the parties with respect to the subject matter hereof. This License shall be governed by the laws of the United States and the State of California, except that body of California law concerning conflicts of law.

Where You are located in the province of Quebec, Canada, the following clause applies: The parties hereby confirm that they have requested that this License and all related documents be drafted in English. Les parties ont exig que le présent contrat et tous les documents connexes soient rédigés en anglais.

EXHIBIT A.

*Portions Copyright (c) 1999-2007 Apple Inc. All Rights Reserved.

This file contains Original Code and/or Modifications of Original Code as defined in and that are subject to the Apple Public Source License Version 2.0 (the 'License'). You may not use this file except in compliance with the License. Please obtain a copy of the License at http://www.opensource.apple.com/apsl/ and read it before using this file.
The Original Code and all software distributed under the License are distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT. Please see the License for the specific language governing rights and limitations under the License."

Copyright 2015 The Android Open Source Project, Inc.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright 2014-2017 Oculus VR, LLC. All rights reserved.

The MIT License (MIT)

Copyright (c) 2013 Greg Thornton

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2006, Google Inc.
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:

//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2009 Tomer Filiba, 2010-2011 Corbin Simpson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-present, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) 2012 Niklas von Hertzen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or their institutions shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the authors.

The SIL OPEN FONT LICENSE applies to the following files:
Gelasio-Bold.ttf
Gelasio-BoldItalic.ttf
Gelasio-Italic.ttf
Gelasio-Regular.ttf
Lohit-Devanagari.ttf
Lohit-Gurmukhi.ttf
Lohit-Tamil.ttf

Copyright 2011-13 Lohit Fonts Project contributors
<http://fedorahosted.org/lohit>

This Font Software is licensed under the SIL Open Font License, Version 1.1. This license is copied below, and is also available with a FAQ at:
http://scripts.sil.org/OFL

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The
requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.
5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

The GPL v2 license applies to the following files
Garuda.ttf
MuktiNarrow.ttf

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

The Vera Bitstream License applies to the following files:
DejaVuSans-Bold.ttf
DejaVuSans.ttf

Fonts are (c) Bitstream (see below). DejaVu changes are in public domain.
Glyphs imported from Arev fonts are (c) Tavmjong Bah (see below)

Bitstream Vera Fonts Copyright

Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is
a trademark of Bitstream, Inc.
Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Bitstream" or the word "Vera".

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Bitstream Vera" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of Gnome, the Gnome Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the Gnome Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

Arev Fonts Copyright
----------------------

Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the fonts accompanying this license ("Fonts") and
associated documentation files (the "Font Software"), to reproduce
and distribute the modifications to the Bitstream Vera Font Software,
including without limitation the rights to use, copy, merge, publish,
distribute, and/or sell copies of the Font Software, and to permit
persons to whom the Font Software is furnished to do so, subject to
the following conditions:

The above copyright and trademark notices and this permission notice
shall be included in all copies of one or more of the Font Software
typefaces.

The Font Software may be modified, altered, or added to, and in
particular the designs of glyphs or characters in the Fonts may be
modified and additional glyphs or characters may be added to the
Fonts, only if the fonts are renamed to names not containing either
the words "Tavmjong Bah" or the word "Arev".

This License becomes null and void to the extent applicable to Fonts
or Font Software that has been modified and is distributed under the
"Tavmjong Bah Arev" names.

The Font Software may be sold as part of a larger software package but
no copy of one or more of the Font Software typefaces may be sold by
itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL
TAVMJONG BAH BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL
DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM
OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the name of Tavmjong Bah shall not
be used in advertising or otherwise to promote the sale, use or other
dealings in this Font Software without prior written authorization
from Tavmjong Bah. For further information, contact: tavmjong @ free
.fr.

TeX Gyre DJV Math

--------------------
Fonts are (c) Bitstream (see below). DejaVu changes are in public domain.

Math extensions done by B. Jackowski, P. Strzelczyk and P. Pianowski
(on behalf of TeX users groups) are in public domain.
Letters imported from Euler Fraktur from AMSfonts are (c) American Mathematical Society (see below).

Bitstream Vera Fonts Copyright
Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license (Fonts) and associated documentation files (the Font Software), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words Bitstream or the word Vera.

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the Bitstream Vera names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.
Except as contained in this notice, the names of GNOME, the GNOME Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the GNOME Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

AMSFonts (v. 2.2) copyright

The PostScript Type 1 implementation of the AMSFonts produced by and previously distributed by Blue Sky Research and Y&Y, Inc. are now freely available for general use. This has been accomplished through the cooperation of a consortium of scientific publishers with Blue Sky Research and Y&Y. Members of this consortium include:

Elsevier Science IBM Corporation Society for Industrial and Applied Mathematics (SIAM) Springer-Verlag American Mathematical Society (AMS)

In order to assure the authenticity of these fonts, copyright will be held by the American Mathematical Society. This is not meant to restrict in any way the legitimate use of the fonts, such as (but not limited to) electronic distribution of documents containing these fonts, inclusion of these fonts into other public domain or commercial font collections or computer applications, use of the outline data to create derivative fonts and/or faces, etc. However, the AMS does require that the AMS copyright notice be removed from any derivative versions of the fonts which have been altered in any way. In addition, to ensure the fidelity of TeX documents using Computer Modern fonts, Professor Donald Knuth, creator of the Computer Modern faces, has requested that any alterations which yield different font metrics be given a different name.

This Font Software is licensed under the SIL Open Font License, Version 1.1.

This license is copied below, and is also available with a FAQ at: http://scripts.sil.org/OFL

The SIL OPEN FONT LICENSE for Noto applies to the following files:
NotoSansKhmer-Regular.ttf

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

------------------------------
Open Source Used In DNAC Platform Arica 6033
PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or
in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

-------------------------------------------------------------------------------------------------------------------------

The Apache License applies to the following files
Arimo-Bold.ttf
Arimo-BoldItalic.ttf
Arimo-Italic.ttf
Arimo-Regular.ttf
Cousine-Bold.ttf
Cousine-BoldItalic.ttf
Cousine-Italic.ttf
Cousine-Regular.ttf
Tinos-Bold.ttf
Tinos-BoldItalic.ttf

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

--------------------------------------------------------------------------------
The public domain/Creative Commons Zero license applies to the following files
Ahem.ttf

The Ahem font in this directory belongs to the public domain. In
jurisdictions that do not recognize public domain ownership of these
files, the following Creative Commons Zero declaration applies:

<http://labs.creativecommons.org/licenses/zero-waive/1.0/us/legalcode>

which is quoted below:

The person who has associated a work with this document (the "Work")
affirms that he or she (the "Affirmer") is the/an author or owner of
the Work. The Work may be any work of authorship, including a
database.

The Affirmer hereby fully, permanently and irrevocably waives and
relinquishes all of her or his copyright and related or neighboring
legal rights in the Work available under any federal or state law,
treaty or contract, including but not limited to moral rights,
publicity and privacy rights, rights protecting against unfair
competition and any rights protecting the extraction, dissemination
and reuse of data, whether such rights are present or future, vested
or contingent (the "Waiver"). The Affirmer makes the Waiver for the
benefit of the public at large and to the detriment of the Affirmer's
heirs or successors.

The Affirmer understands and intends that the Waiver has the effect
of eliminating and entirely removing from the Affirmer's control all
the copyright and related or neighboring legal rights previously held
by the Affirmer in the Work, to that extent making the Work freely
available to the public for any and all uses and purposes without
restriction of any kind, including commercial use and uses in media
and formats or by methods that have not yet been invented or
conceived. Should the Waiver for any reason be judged legally
ineffective in any jurisdiction, the Affirmer hereby grants a free,
full, permanent, irrevocable, nonexclusive and worldwide license for
all her or his copyright and related or neighboring legal rights in
the Work.

The public domain license applies to the following files
GardinerModBug.ttf
GardinerModCat.ttf

In lieu of a licence Fonts in this site are offered free for any use; they may be installed, embedded, opened, edited,
modified, regenerated, posted, packaged and redistributed. George Douros

**QT LICENSE AGREEMENT**
Agreement version 4.2.1

This Qt License Agreement ("Agreement") is a legal agreement for the licensing
of Licensed Software (as defined below) between The Qt Company (as defined
below) and the Licensee who has accepted the terms of this Agreement by
downloading or using the Licensed Software and/or as defined herein:

Capitalized terms used herein are defined in Section 1.

**WHEREAS:**
(A) Licensee wishes to use the Licensed Software for the purpose of
developing and distributing Applications and/or Devices (each as
defined below); and
(B) The Qt Company is willing to grant the Licensee a right to use
Licensed Software for such a purpose pursuant to term and conditions
of this Agreement.

**NOW, THEREFORE, THE PARTIES HEREBY AGREE AS FOLLOWS:**

1. **DEFINITIONS**
"Affiliate” of a Party shall mean an entity (i) which is directly or indirectly
controlling such Party; (ii) which is under the same direct or indirect
ownership or control as such Party; or (iii) which is directly or indirectly
owned or controlled by such Party. For these purposes, an entity shall be
treated as being controlled by another if that other entity has fifty percent
(50 %) or more of the votes in such entity, is able to direct its affairs
and/or to control the composition of its board of directors or equivalent body.

"Add-on Products” shall mean The Qt Company's specific add-on software products
(for example Qt Safe Renderer, Qt for Automation, Qt Application Manager),
which are not licensed as part of The Qt Company's standard offering, but shall
be included into the scope of Licensed Software only if so specifically agreed
between the Parties.

"Applications" shall mean Licensee's software products created using the Licensed Software, which may include the Redistributables, or part thereof.

"Contractor(s)" shall mean third party consultants, distributors and contractors performing services to the Licensee under applicable contractual arrangement.

"Customer(s)" shall mean Licensee's end users to whom Licensee, directly or indirectly, distributes copies of the Redistributables.

"Data Protection Legislation" shall mean the General Data Protection Regulation (EU 2016/679) (GDPR) and any national implementing laws, regulations and secondary legislation, as may be amended or updated from time to time, as well as any other data protection laws or regulations applicable in relevant territory.

"Deployment Platforms" shall mean operating systems and/or hardware specified in the License Certificate, on which the Redistributables can be distributed pursuant to the terms and conditions of this Agreement.

"Designated User(s)" shall mean the employee(s) of Licensee or Licensee's Affiliates acting within the scope of their employment or Licensee's Contractors acting within the scope of their services for Licensee and on behalf of Licensee. Designated Users shall be named in the License Certificate.

"Development License" shall mean the license needed by the Licensee for each Designated User to use the Licensed Software under the license grant described in Section 3.1 of this Agreement. Development Licenses are available separately for Qt for Application Development and Qt for Device Creation products, each product having its designated scope and purpose of use. Distribution Licenses are always connected to Qt for Device Creation product only.

"Development Platforms" shall mean those operating systems specified in the License Certificate, in which the Licensed Software can be used under the Development License, but not distributed in any form or used for any other purpose.

"Devices" shall mean hardware devices or products that 1) are manufactured and/or distributed by the Licensee or its Affiliates or Contractors, and (2)(i) incorporate or integrate the Redistributables or parts thereof; or (ii) where the main user interface or substantial functionality of such unit, when used by a Customer, is provided by Application(s) or otherwise depends on the Licensed Software, regardless of whether the Redistributables are distributed together with the hardware or not. Devices covered with this Agreement shall be specified in Appendix 2 or in a quote.
"Distribution License(s)" shall mean the license required for any kind of sale, trade, exchange, loan, lease, rental or other distribution by or on behalf of Licensee to a third party of Redistributables in connection with Devices pursuant to license grant described in Section 3.3 of this Agreement.

"Distribution License Packs" shall mean set of prepaid Distribution Licenses for distribution of Redistributables, as defined in The Qt Company's standard price list, quote, Purchase Order confirmation or in an appendix hereto, as the case may be.

"Intellectual Property Rights" shall mean patents (including utility models), design patents, and designs (whether or not capable of registration), chip topography rights and other like protection, copyrights, trademarks, service marks, trade names, logos or other words or symbols and any other form of statutory protection of any kind and applications for any of the foregoing as well as any trade secrets.

"License Certificate" shall mean a certificate generated by The Qt Company for each Designated User respectively upon them downloading the Licensed Software, which will be available under respective Designated User's Qt Account at account.qt.io. License Certificates will specify the Designated User, the Development Platforms, Deployment Platforms and the License Term. Such terms are considered part of the licenses granted hereunder and shall be updated from time to time to reflect any agreed changes to the foregoing terms relating to Designated User's rights to the Licensed Software.

"License Fee" shall mean the fee charged to the Licensee for rights granted under the terms of this Agreement.

"License Term" shall mean the agreed validity period of the Development License of the respective Designated User, during which time the Designated User is entitled to use the Licensed Software, as set forth in the respective License Certificate.

"Licensed Software" shall mean either
(i) Qt for Application Development or
(ii) Qt for Device Creation, and/or
(iii) Qt 3D Studio, and/or
(iv) Qt Design Studio, and/or
(v) Qt for MCUs, and/or
(vi) selected Add-on Products, if any,

depending on which product(s) the Licensee has purchased under this Agreement, as well as corresponding online or electronic documentation, associated media and printed materials, including the source code (where applicable), example programs and the documentation, licensed to the Licensee under this Agreement. Licensed Software does not include Third Party Software (as defined in Section
4) or Open Source Qt. The Qt Company may, in the course of its development activities, at its free and absolute discretion and without any obligation to send or publish any notifications to the Licensee or in general, make changes, additions or deletions in the components and functionalities of the Licensed Software, provided that no such changes, additions or deletions will affect the already released version of the Licensed Software, but only upcoming version(s).

"Licensee" shall mean the individual or legal entity that is party to this Agreement, as identified on the signature page hereof.

"Licensee's Records" shall mean books and records that are likely to contain information bearing on Licensee's compliance with this Agreement, Licensee's use of Open Source Qt and/or the payments due to The Qt Company under this Agreement, including, but not limited to user information, assembly logs, sales records and distribution records.

"Modified Software" shall have the meaning as set forth in Section 2.3.

"Online Services" shall mean any services or access to systems made available by The Qt Company to the Licensee over the Internet relating to the Licensed Software or for the purpose of use by the Licensee of the Licensed Software or Support. Use of any such Online Services is discretionary for the Licensee and some of them may be subject to additional fees.

"Open Source Qt" shall mean the non-commercial Qt computer software products, licensed under the terms of the GNU Lesser General Public License, version 2.1 or later ("LGPL") or the GNU General Public License, version 2.0 or later ("GPL"). For clarity, Open Source Qt shall not be provided nor governed under this Agreement.

"Party" or "Parties" shall mean Licensee and/or The Qt Company.

"Permitted Combination" shall have the meaning as set forth in Section 3.4(viii).

"Pre-Release Code" shall have the meaning as set forth in Section 4.

"Prohibited Combination" shall mean any means to (i) use, combine, incorporate, link or integrate Licensed Software with any software created with or incorporating Open Source Qt, (ii) use Licensed Software for creation of any software created with or incorporating Open Source Qt, or (iii) incorporate or integrate Applications into a hardware device or product other than a Device.

"Qt 3D Studio" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt Design Studio" shall mean The Qt Company's productized offering, which
consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for Application Development" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for Device Creation" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Qt for MCUs" shall mean The Qt Company's productized offering, which consist of all versions of modules and tools as set forth in Appendix 1.

"Redistributables" shall mean the portions of the Licensed Software set forth in Appendix 1 that may be distributed pursuant to the terms of this Agreement in object code form only, including any relevant documentation. Where relevant, any reference to Licensed Software in this Agreement shall include and refer also to Redistributables.

"Renewal Term" shall mean an extension of previous License Term as agreed between the Parties.

"Submitted Modified Software" shall have the meaning as set forth in Section 2.3.

"Support" shall mean standard developer support that is provided by The Qt Company to assist Designated Users in using the Licensed Software in accordance with The Qt Company's standard support terms available at https://www.qt.io/terms-conditions/ and as further defined in Section 8 hereunder.

"Taxes" shall have the meaning set forth in Section 10.5.

"Term" shall have the meaning set forth in Section 12.

"The Qt Company" shall mean:

(i) in the event Licensee is an individual residing in the United States or a legal entity incorporated in the United States or having its headquarters in the United States, The Qt Company Inc., a Delaware corporation with its office at 2350 Mission College Blvd., Suite 1020, Santa Clara, CA 95054, USA.; or

(ii) in the event the Licensee is an individual residing outside of the United States or a legal entity incorporated outside of the United States or having its registered office outside of the United States, The Qt Company Ltd., a Finnish company with its registered office at Bertel Jungin aukio D3A, 02600 Espoo, Finland.

"Third-Party Software" shall have the meaning set forth in Section 4.
"Updates" shall mean a release or version of the Licensed Software containing bug fixes, error corrections and other changes that are generally made available to users of the Licensed Software that have contracted for Support. Updates are generally depicted as a change to the digits following the decimal in the Licensed Software version number. The Qt Company shall make Updates available to the Licensee under the Support. Updates shall be considered as part of the Licensed Software hereunder.

"Upgrades" shall mean a release or version of the Licensed Software containing enhancements and new features and are generally depicted as a change to the first digit of the Licensed Software version number. In the event Upgrades are provided to the Licensee under this Agreement, they shall be considered as part of the Licensed Software hereunder.

2. OWNERSHIP

2.1. Ownership of The Qt Company

The Licensed Software is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Licensed Software is licensed, not sold.

All of The Qt Company’s Intellectual Property Rights are and shall remain the exclusive property of The Qt Company or its licensors respectively.

2.2. Ownership of Licensee

All the Licensee’s Intellectual Property Rights are and shall remain the exclusive property of the Licensee or its licensors respectively.

All Intellectual Property Rights to the Modified Software, Applications and Devices shall remain with the Licensee and no rights thereto shall be granted by the Licensee to The Qt Company under this Agreement (except as set forth in Section 2.3 below).

2.3. Modified Software

Licensee may create bug-fixes, error corrections, patches or modifications to the Licensed Software (“Modified Software”). Such Modified Software may break the source or binary compatibility with the Licensed Software (including without limitation through changing the application programming interfaces (“API”) or by adding, changing or deleting any variable, method, or class signature in the Licensed Software and/or any inter-process protocols, services or standards in the Licensed Software libraries). To the extent that Licensee’s Modified Software so breaks source or binary compatibility with the Licensed Software, Licensee acknowledges that The Qt Company’s ability to provide Support may be prevented or limited and Licensee’s ability to make use of Updates may be restricted.

Licensee may, at its sole and absolute discretion, choose to submit Modified Software to The Qt Company (“Submitted Modified Software”) in connection with
Licensee's Support request, service request or otherwise. In the event Licensee does so, then, Licensee hereby grants The Qt Company a sublicensable, assignable, irrevocable, perpetual, worldwide, non-exclusive, royalty-free and fully paid-up license, under all of Licensee's Intellectual Property Rights, to reproduce, adapt, translate, modify, and prepare derivative works of, publicly display, publicly perform, sublicense, make available and distribute such Submitted Modified Software as The Qt Company sees fit at its free and absolute discretion.

3. LICENSES GRANTED
3.1. Development with Licensed Software
Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable license, valid for the License Term, to use, modify and copy the Licensed Software by Designated Users on the Development Platforms for the sole purposes of designing, developing, demonstrating and testing Application(s) and/or Devices, and to provide thereto related support and other related services to end-user Customers.

Licensee may install copies of the Licensed Software on five (5) computers per Designated User, provided that only the Designated Users who have a valid Development License may use the Licensed Software.

Licensee may at any time designate another Designated User to replace a then-current Designated User by notifying The Qt Company in writing, provided that any Designated User may be replaced only once during any six-month period.

Upon expiry of the initially agreed License Term, the respective License Terms shall be automatically extended to one or more Renewal Term(s), unless and until either Party notifies the other Party in writing that it does not wish to continue the License Term, such notification to be provided to the other Party no less than ninety (90) days before expiry of the respective License Term. Unless otherwise agreed between the Parties, Renewal Term shall be of equal length with the initial License Term.

Any such Renewal Term shall be subject to License Fees agreed between the Parties or, if no advance agreement exists, subject to The Qt Company's standard pricing applicable at the commencement date of any such Renewal Term.

Any price or other term specified for a Renewal Term shall be valid only for the specified time.

The Qt Company may request the Licensee to place a purchase order corresponding to a quote by The Qt Company for the relevant Renewal Term.

In the event Licensee does not prevent auto-renewal pursuant the above, but a Renewal Term is nevertheless not duly ordered within 30 days from the date of the respective quote from The Qt Company and/or the respective License Fee paid by due date specified in The Qt Company's respective invoice, The Qt
Company shall apply a reinstatement fee equal to ten percent (10 %) of the total value of the License Fees of the Development Licenses for the expired term to be added to the License Fee of the respective Renewal Term.

In the event Licensee chooses not to renew a Development License for a Renewal Term by notifying The Qt Company thereof no less than ninety (90) days before expiry of the respective License Term, Licensee may still reinstate such expired Development Licenses for a Renewal Term subject to applicable renewal Term License Fees until thirty (30) days from the expiry of the initially agreed License Term or preceding Renewal Term. After such thirty (30) day period a Development License shall be subject to applicable License Fees for a new Development License and not any Renewal Term License Fees.

3.2. Distribution of Applications

Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable, revocable (for cause pursuant to this Agreement) right and license, valid for the Term, to

(i) distribute, by itself or through its Contractors, Redistributables as installed, incorporated or integrated into Applications for execution on the Deployment Platforms, and

(ii) grant sublicenses to Redistributables, as distributed hereunder, for Customers solely for Customer's internal use and to the extent necessary in order for the Customers to use the Applications for their respective intended purposes.

Right to distribute the Redistributables as part of an Application as provided herein is not royalty-bearing but is conditional upon the Licensee not having any unpaid License Fees for Development Licenses owed to The Qt Company at the time of distribution of any Redistributables to Customers.

3.3. Distribution of Devices

Subject to the terms of this Agreement, The Qt Company grants to Licensee a worldwide, non-exclusive, non-transferable, revocable (for cause pursuant to this Agreement) right and license, valid for the Term, to

(i) distribute, by itself or through one or more tiers of Contractors, Redistributables as installed, incorporated or integrated, or intended to be installed, incorporated or integrated into Devices for execution on the Deployment Platforms, and

(ii) grant sublicenses to Redistributables, as distributed hereunder, for Customers solely for Customer's internal use and to the extent necessary in order for the Customers to use the Devices for their respective intended purposes.

Right to distribute the Redistributables with Devices as provided herein is conditional upon the Licensee (i) not having any unpaid License Fees for Development Licenses owed to The Qt Company, and (ii) having purchased and paid corresponding Distribution Licenses at the time of distribution of any Redistributables to Customers.
3.4. Further Requirements
The licenses granted above in this Section 3 by The Qt Company to Licensee are conditional and subject to Licensee's compliance with the following terms:

(i) Licensee shall not remove or alter any copyright, trademark or other proprietary rights notice(s) contained in any portion of the Licensed Software;

(ii) Applications must add primary and substantial functionality to the Licensed Software so as not to compete with theLicensed Software;

(iii) Applications may not pass on functionality which in any way makes it possible for others to create software with the Licensed Software; provided however that Licensee may use the Licensed Software's scripting and QML ("Qt Quick") functionality solely in order to enable scripting, themes and styles that augment the functionality and appearance of the Application(s) without adding primary and substantial functionality to the Application(s);

(iv) Licensee shall not use Licensed Software in any manner or for any purpose that infringes, misappropriates or otherwise violates any Intellectual property or right of any third party, or that violates any applicable law;

(v) Licensee shall not use The Qt Company's or any of its suppliers' names, logos, or trademarks to market Applications, except that Licensee may use "Built with Qt" logo to indicate that Application(s) was developed using the Licensed Software;

(vi) Licensee shall not distribute, sublicense or disclose source code of Licensed Software to any third party (provided however that Licensee may appoint employee(s) of Contractors as Designated Users to use Licensed Software pursuant to this Agreement). Such right may be available for the Licensee subject to a separate software development kit ("SDK") license agreement to be concluded with The Qt Company;

(vii) Licensee shall not grant the Customers a right to (i) make copies of the Redistributables except when and to the extent required to use the Applications and/or Devices for their intended purpose, (ii) modify the Redistributables or create derivative works thereof, (iii) decompile, disassemble or otherwise reverse engineer Redistributables, or (iv) redistribute any copy or portion of the Redistributables to any third party, except as part of the onward sale of the Device on which the Redistributables are installed;

(viii) Licensee shall not and shall cause that its Affiliates or Contractors shall not use Licensed Software in any Prohibited Combination, unless Licensee has received an advance written permission from The Qt Company to do so. Absent such written permission, any and all distribution by the Licensee during the Term of a hardware device or product a) which incorporate or integrate any part of Licensed Software or Open Source Qt; or b) where the main user interface or substantial functionality is
provided by software built with Licensed Software or Open Source Qt or otherwise depends on the Licensed Software or Open Source Qt, shall be considered to be Device distribution under this Agreement and shall be dependent on Licensee's compliance thereof (including but not limited to obligation to pay applicable License Fees for such distribution). Notwithstanding what is provided above in this sub-section (viii), Licensee is entitled to use and combine Qt 3D Studio and/or Qt Design Studio with Open Source Qt ("Permitted Combination") for its internal evaluation purposes, provided that Licensee shall in no way transfer, publish, disclose, display or otherwise make available any software or work resulting from such Permitted Combination;

(ix) Licensee shall cause all of its Affiliates and Contractors entitled to make use of the licenses granted under this Agreement, to be contractually bound to comply with the relevant terms of this Agreement and not to use the Licensed Software beyond the terms hereof and for any purposes other than operating within the scope of their services for Licensee. Licensee shall be responsible for any and all actions and omissions of its Affiliates and Contractors relating to the Licensed Software and use thereof (including but not limited to payment of all applicable License Fees);

(x) Except when and to the extent explicitly provided in this Section 3, Licensee shall not transfer, publish, disclose, display or otherwise make available the Licensed Software; and

(xi) Licensee shall not attempt or enlist a third party to conduct or attempt to conduct any of the above.

Above terms shall not be applicable if and to the extent they conflict with any mandatory provisions of any applicable laws.

Any use of Licensed Software beyond the provisions of this Agreement is strictly prohibited and requires an additional license from The Qt Company.

4. THIRD-PARTY SOFTWARE

The Licensed Software may provide links to third party libraries or code (collectively "Third-Party Software") to implement various functions. Third-Party Software does not comprise part of the Licensed Software. In some cases, access to Third-Party Software may be included with the Licensed Software. Such Third-Party Software will be listed in the "../src/3rdparty" source tree delivered with the Licensed Software or documented in the Licensed Software, as such may be amended from time to time. Licensee acknowledges that use or distribution of Third-Party Software is in all respects subject to applicable license terms of applicable third-party right holders.

5. PRE-RELEASE CODE

The Licensed Software may contain pre-release code and functionality, or sample code marked or otherwise stated with appropriate designation such as "Technology Preview", "Alpha", "Beta", "Sample" etc. ("Pre-Release Code").
Such Pre-Release Code may be present complimentary for the Licensee, in order to provide experimental support or information for new platforms or preliminary versions of one or more new functionalities or for other similar reasons. The Pre-Release Code may not be at the level of performance and compatibility of a final, generally available, product offering. The Pre-Release Code may not operate correctly, may contain errors and may be substantially modified by The Qt Company prior to the first commercial product release, if any. The Qt Company is under no obligation to make Pre-Release Code commercially available, or provide any Support or Updates relating thereto. The Qt Company assumes no liability whatsoever regarding any Pre-Release Code, but any use thereof is exclusively at Licensee’s own risk and expense.

For clarity, unless Licensed Software specifies different license terms for the respective Pre-Release Code, the Licensee is entitled to use such pre-release code pursuant to Section 3, just like other Licensed Software, provided however that in the event Add-on Products are included and available as such Pre-Release Code, Licensee’s right to use such Add-on Products is nevertheless subject to and conditional upon conclusion of separate agreement with The Qt Company.

6. LIMITED WARRANTY AND WARRANTY DISCLAIMER
The Qt Company hereby represents and warrants that it has the power and authority to grant the rights and licenses granted to Licensee under this Agreement.

Except as set forth above, the Licensed Software is licensed to Licensee "as is" and Licensee’s exclusive remedy and The Qt Company’s entire liability for errors in the Licensed Software shall be limited, at The Qt Company’s option, to correction of the error, replacement of the Licensed Software or return of the applicable fees paid for the defective Licensed Software for the time period during which the License is not able to utilize the Licensed Software under the terms of this Agreement.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE QT COMPANY ON BEHALF OF ITSELF AND ITS LICENSORS, SUPPLIERS AND AFFILIATES, DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT WITH REGARD TO THE LICENSED SOFTWARE. THE QT COMPANY DOES NOT WARRANT THAT THE LICENSED SOFTWARE WILL SATISFY LICENSEE’S REQUIREMENTS OR THAT IT WILL OPERATE WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED.

7. INDEMNIFICATION AND LIMITATION OF LIABILITY
7.1. Limitation of Liability
EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, AND (II) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY BE LIABLE TO THE OTHER PARTY FOR ANY LOSS OF PROFIT,
LOSS OF DATA, LOSS OF BUSINESS OR GOODWILL OR ANY OTHER INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE COST, DAMAGES OR EXPENSE OF ANY KIND, HOWSOEVER ARISING UNDER OR IN CONNECTION WITH THIS AGREEMENT.

EXCEPT FOR (I) CASES OF GROSS NEGLIGENCE OR INTENTIONAL MISCONDUCT, AND (II) BREACH OF CONFIDENTIALITY, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL EITHER PARTY’S TOTAL AGGREGATE LIABILITY UNDER THIS AGREEMENT EXCEED THE AGGREGATE LICENSE FEES PAID OR PAYABLE TO THE QT COMPANY FROM LICENSEE DURING THE PERIOD OF TWELVE (12) MONTHS IMMEDIATELY PRECEDING THE EVENT RESULTING IN SUCH LIABILITY.

THE PROVISIONS OF THIS SECTION 7 ALLOCATE THE RISKS UNDER THIS AGREEMENT BETWEEN THE QT COMPANY AND LICENSEE AND THE PARTIES HAVE RELIED UPON THE LIMITATIONS SET FORTH HEREIN IN DETERMINING WHETHER TO ENTER INTO THIS AGREEMENT.

NOTWITHSTANDING ANYTHING TO THE CONTRARY IN THIS AGREEMENT, LICENSEE SHALL ALWAYS BE LIABLE TO PAY THE APPLICABLE LICENSE FEES CORRESPONDING TO ITS ACTUAL USE OF LICENSED SOFTWARE.

8. SUPPORT, UPDATES AND ONLINE SERVICES

Upon due payment of the agreed License Fees the Licensee will be eligible to receive Support and Updates and to use the Online Services during the License Term, provided, however, that in the event the License Term is longer than 36 months, the initial payment includes Support for only the first 12 months, unless the Parties specifically otherwise agree.

Unless otherwise decided by The Company at its free and absolute discretion, Upgrades will not be included in the Support but may be available subject to additional fees.

From time to time The Qt Company may change the Support terms, provided that during the respective ongoing License Term the level of Support provided by The Qt Company may not be reduced without the consent of the Licensee.

Unless otherwise agreed, The Qt Company shall not be responsible for providing any service or support to Customers.

9. CONFIDENTIALITY

Each Party acknowledges that during the Term of this Agreement each Party may receive information about the other Party’s business, business methods, business plans, customers, business relations, technology, and other information, including the terms of this Agreement, that is confidential and of great value to the other Party, and the value of which would be significantly reduced if disclosed to third parties ("Confidential Information"). Accordingly, when a Party (the "Receiving Party") receives Confidential Information from the other Party (the "Disclosing Party"), the Receiving Party shall only disclose such information to employees and Contractors on a need to
know basis, and shall cause its employees and employees of its Affiliates to:
(i) maintain any and all Confidential Information in confidence; (ii) not
disclose the Confidential Information to a third party without the Disclosing
Party's prior written approval; and (iii) not, directly or indirectly, use the
Confidential Information for any purpose other than for exercising its rights
and fulfilling its responsibilities pursuant to this Agreement. Each Party
shall take reasonable measures to protect the Confidential Information of the
other Party, which measures shall not be less than the measures taken by such
Party to protect its own confidential and proprietary information.

Obligation of confidentiality shall not apply to information that (i) is or
becomes generally known to the public through no act or omission of the
Receiving Party; (ii) was in the Receiving Party's lawful possession prior to
the disclosure hereunder and was not subject to limitations on disclosure or
use; (iii) is developed independently by employees or Contractors of the
Receiving Party or other persons working for the Receiving Party who have not
had access to the Confidential Information of the Disclosing Party, as proven
by the written records of the Receiving Party; (iv) is lawfully disclosed to
the Receiving Party without restrictions, by a third party not under an
obligation of confidentiality; or (v) the Receiving Party is legally compelled
to disclose, in which case the Receiving Party shall notify the Disclosing
Party of such compelled disclosure and assert the privileged and confidential
nature of the information and cooperate fully with the Disclosing Party to
limit the scope of disclosure and the dissemination of disclosed Confidential
Information to the minimum extent necessary.

The obligations under this Section 9 shall continue to remain in force for a
period of five (5) years after the last disclosure, and, with respect to trade
secrets, for so long as such trade secrets are protected under applicable trade
secret laws.

10. FEES, DELIVERY AND PAYMENT

10.1. License Fees
License Fees are described in The Qt Company's standard price list, quote or
Purchase Order confirmation or in an appendix hereto, as the case may be.

The License Fees shall not be refunded or claimed as a credit in any event or
for any reason whatsoever.

10.2. Ordering Licenses
Licensee may purchase Development Licenses and Distribution Licenses pursuant
to agreed pricing terms or, if no specific pricing terms have been agreed upon,
at The Qt Company's standard pricing terms applicable at the time of purchase.
Unless specifically otherwise provided, any pricing terms referenced in this
Agreement shall be valid for twelve (12) months from the date of this Agreement.

Licensee shall submit all purchase orders for Development Licenses and
Distribution Licenses to The Qt Company by email or any other method acceptable
to The Qt Company (each such order is referred to herein as a “Purchase Order”) for confirmation, whereupon the Purchase Order shall become binding between the Parties.

10.3. Distribution License Packs
Unless otherwise agreed, Distribution Licenses shall be purchased by way of Distribution License Packs.

Upon due payment of the ordered Distribution License Pack(s), the Licensee will have an account of Distribution Licenses available for distributing the Redistributables in accordance with this Agreement.

Each time Licensee distributes a copy of Redistributables, then one Distribution License is used, and Licensee's account of available Distribution Licenses is decreased accordingly. Licensee may distribute copies of the Redistributables so long as Licensee has Distribution Licenses remaining on its account.

10.4. Payment Terms
License Fees and any other charges under this Agreement shall be paid by Licensee no later than thirty (30) days from the date of the applicable invoice from The Qt Company.

The Qt Company will submit an invoice to Licensee after the date of this Agreement and/or after The Qt Company receives a Purchase Order from Licensee.

A late payment charge of the lower of (a) one percent per month; or (b) the interest rate stipulated by applicable law, shall be charged on any unpaid balances that remain past due.

10.5. Taxes
All License Fees and other charges payable hereunder are gross amounts but exclusive of any value added tax, use tax, sales tax, withholding tax and other taxes, duties or tariffs (“Taxes”) levied directly for the sale, delivery or use of Licensed Software hereunder pursuant to any applicable law. Such applicable Taxes shall be paid by Licensee to The Qt Company, or, where applicable, in lieu of payment of such Taxes to The Qt Company, Licensee shall provide an exemption certificate to The Qt Company and any applicable authority.

11. RECORD-KEEPING AND REPORTING OBLIGATIONS; AUDIT RIGHTS
11.1. Licensee's Record-keeping
Licensee shall at all times during the Term of this Agreement and for a period of seven (7) years thereafter maintain Licensee's Records in an accurate and up-to-date form. Licensee's Records shall be adequate to reasonably enable The Qt Company to determine Licensee's compliance with the provisions of this Agreement. The records shall conform to general good accounting practices.

Licensee shall, within thirty (30) days from receiving The Qt Company’s request
to that effect, deliver to The Qt Company a report based on Licensee's Records, such report to contain information, in sufficient detail, on (i) number and identity of users working with Licensed Software or Open Source Qt, (ii) copies of Redistributables distributed by Licensee during the most recent calendar quarter and/or any other term specified by The Qt Company, (iii) number of undistributed copies of Redistributables and corresponding number of unused Distribution Licenses remaining on Licensee's account, and (iv) any other information as The Qt Company may reasonably require from time to time.

11.2. The Qt Company's Audit Rights
The Qt Company or an independent auditor acting on behalf of The Qt Company, may, upon at least five (5) business days' prior written notice and at its expense, audit Licensee with respect to the Licensee's use of the Licensed Software, but not more frequently than once during each 6-month period. Such audit may be conducted by mail, electronic means or through an in-person visit to Licensee's place of business. Any such in-person audit shall be conducted during regular business hours at Licensee's facilities and shall not unreasonably interfere with Licensee's business activities. The Qt Company or the independent auditor acting on behalf of The Qt Company shall be entitled to inspect Licensee's Records and conduct necessary interviews of Licensee's relevant employees and Contractors. All such Licensee's Records and use thereof shall be subject to an obligation of confidentiality under this Agreement.

If an audit reveals that Licensee is using the Licensed Software beyond scope of the licenses Licensee has paid for, Licensee agrees to pay The Qt Company any amounts owed for such unauthorized use within 30 days from receipt of the corresponding invoice from The Qt Company. In addition, in the event the audit reveals a material violation of the terms of this Agreement (without limitation, either (i) underpayment of more than 10% of License Fees or 10,000 euros (whichever is more) or (ii) distribution of products, which include or result from Prohibited Combination, shall be deemed a material violation for purposes of this section), then the Licensee shall pay The Qt Company's reasonable cost of conducting such audit.

12. TERM AND TERMINATION
12.1. Agreement Term
This Agreement shall enter into force upon due acceptance by both Parties and remain in force for as long as there is any Development License(s) purchased under this Agreement in force ("Term"), unless and until terminated pursuant to the terms of this Section 12.

12.2. Termination and suspension of rights
Either Party shall have the right to terminate this Agreement upon thirty (30) days prior written notice if the other Party commits a material breach of any obligation of this Agreement and fails to remedy such breach within such notice period.

Instead of termination, The Qt Company shall have the right to suspend or
withhold grants of all rights to the Licensed Software hereunder, including but not limited to the Development Licenses, Distribution License, and Support, should Licensee fail to make payment in timely fashion or otherwise violates or is reasonably suspected to violate its obligations or terms of this Agreement, and where such violation or breach is not cured within five (5) business days following The Qt Company's written notice thereof.

12.3. Mutual Right to Terminate
Either Party shall have the right to terminate this Agreement immediately upon written notice in the event that the other Party becomes insolvent, files for any form of bankruptcy, makes any assignment for the benefit of creditors, has a receiver, administrative receiver or officer appointed over the whole or a substantial part of its assets, ceases to conduct business, or an act equivalent to any of the above occurs under the laws of the jurisdiction of the other Party.

12.4. Parties Rights and Duties upon Termination
Upon expiry or termination of the Agreement, Licensee shall cease and shall cause all Designated Users (including those of its Affiliates' and Contractors') to cease using the Licensed Software and distribution of the Redistributables under this Agreement.

Notwithstanding the above, in the event the Agreement expires or is terminated:
(i) as a result of The Qt Company choosing not to renew the Development License(s) as set forth in Section 3.1, then all valid licenses possessed by the Licensee at such date shall be extended to be valid in perpetuity under the terms of this Agreement and Licensee is entitled to purchase additional licenses as set forth in Section 10.2; or
(ii) for reason other than by The Qt Company pursuant to item (i) above or pursuant to Section 12.2, then the Licensee is entitled, for a period of six (6) months after the effective date of termination, to continue distribution of Devices under the Distribution Licenses paid but unused at such effective date of termination.

Upon any such termination the Licensee shall destroy or return to The Qt Company all copies of the Licensed Software and all related materials and will certify the same to The Qt Company upon its request, provided however that Licensee may retain and exploit such copies of the Licensed Software as it may reasonably require in providing continued support to Customers.

Expiry or termination of this Agreement for any reason whatsoever shall not relieve Licensee of its obligation to pay any License Fees accrued or payable to The Qt Company prior to the effective date of termination, and Licensee shall immediately pay to The Qt Company all such fees upon the effective date of termination. Termination of this Agreement shall not affect any rights of Customers to continue use of Applications and Devices (and therein incorporated Redistributables).
12.5. Extension in case of bankruptcy
In the event The Qt Company is declared bankrupt under a final, non-cancellable
decision by relevant court of law, and this Agreement is not, at the date of
expiry of the Development License(s) pursuant to Section 3.1, assigned to
party, who has assumed The Qt Company's position as a legitimate licensor of
Licensed Software under this Agreement, then all valid licenses possessed by
the Licensee at such date of expiry, and which the Licensee has not notified
for expiry, shall be extended to be valid in perpetuity under the terms of this
Agreement.

13. GOVERNING LAW AND LEGAL VENUE
In the event this Agreement is in the name of The Qt Company Inc., a Delaware
Corporation, then:
   (i)  this Agreement shall be construed and interpreted in accordance
        with the laws of the State of California, USA, excluding its choice
        of law provisions;
   (ii) the United Nations Convention on Contracts for the International
        Sale of Goods will not apply to this Agreement; and
   (iii) any dispute, claim or controversy arising out of or relating to
        this Agreement or the breach, termination, enforcement,
        interpretation or validity thereof, including the determination of
        the scope or applicability of this Agreement to arbitrate, shall
        be determined by arbitration in San Francisco, USA, before one
        arbitrator. The arbitration shall be administered by JAMS pursuant
        to JAMS' Streamlined Arbitration Rules and Procedures. Judgment on
        the Award may be entered in any court having jurisdiction. This
        Section shall not preclude parties from seeking provisional
        remedies in aid of arbitration from a court of appropriate
        jurisdiction.

In the event this Agreement is in the name of The Qt Company Ltd., a Finnish
Company, then:
   (i)  this Agreement shall be construed and interpreted in accordance
        with the laws of Finland, excluding its choice of law provisions;
   (ii) the United Nations Convention on Contracts for the International
        Sale of Goods will not apply to this Agreement; and
   (iii) any disputes, controversy or claim arising out of or relating to
        this Agreement, or the breach, termination or validity thereof
        shall be finally settled by arbitration in accordance with the
        Arbitration Rules of Finland Chamber of Commerce. The arbitration
        tribunal shall consist of one (1), or if either Party so requires,
        of three (3), arbitrators. The award shall be final and binding and
        enforceable in any court of competent jurisdiction. The arbitration
        shall be held in Helsinki, Finland and the process shall be
        conducted in the English language. This Section shall not preclude
        parties from seeking provisional remedies in aid of arbitration
        from a court of appropriate jurisdiction.
14. GENERAL PROVISIONS

14.1. No Assignment
Except in the case of a merger or sale of substantially all of its corporate assets, Licensee shall not be entitled to assign or transfer all or any of its rights, benefits and obligations under this Agreement without the prior written consent of The Qt Company, which shall not be unreasonably withheld or delayed. The Qt Company shall be entitled to freely assign or transfer any of its rights, benefits or obligations under this Agreement.

14.2. No Third-Party Representations
Licensee shall make no representations or warranties concerning the Licensed Software on behalf of The Qt Company. Any representation or warranty Licensee makes or purports to make on The Qt Company's behalf shall be void as to The Qt Company.

14.3. Surviving Sections
Any terms and conditions that by their nature or otherwise reasonably should survive termination of this Agreement shall so be deemed to survive. Such sections include especially the following: 1, 2, 6, 7, 9, 11, 12.4, 13 and 14.

14.4. Entire Agreement
This Agreement, the exhibits hereto, the License Certificate and any applicable Purchase Order accepted by The Qt Company constitute the complete agreement between the Parties and supersedes all prior or contemporaneous discussions, representations, and proposals, written or oral, with respect to the subject matters discussed herein.

In the event of any conflict or inconsistency between this Agreement and any Purchase Order, the terms of this Agreement will prevail over the terms of the Purchase Order with respect to such conflict or inconsistency.

Parties specifically acknowledge and agree that this Agreement prevails over any click-to-accept or similar agreements the Designated Users may need to accept online upon download of the Licensed Software, as may be required by The Qt Company's applicable processes relating to Licensed Software.

14.5. Modifications
No modification of this Agreement shall be effective unless contained in a writing executed by an authorized representative of each Party. No term or condition contained in Licensee's Purchase Order ("Deviating Terms") shall apply unless The Qt Company has expressly agreed such Deviating Terms in writing. Unless and to the extent expressly agreed by The Qt Company, any such Deviating Terms shall be deemed void and with no legal effect. For clarity, delivery of the Licensed Software following the receipt of the Purchase Order including Deviating Terms shall not constitute acceptance of such Deviating Terms."
14.6. Force Majeure
Except for the payment obligations hereunder, neither Party shall be liable to
the other for any delay or non-performance of its obligations hereunder in the
event and to the extent that such delay or non-performance is due to an event
of act of God, terrorist attack or other similar unforeseeable catastrophic
event that prevents either Party for fulfilling its obligations under this
Agreement and which such Party cannot avoid or circumvent ("Force Majeure
Event"). If the Force Majeure Event results in a delay or non-performance of a
Party for a period of three (3) months or longer, then either Party shall have
the right to terminate this Agreement with immediate effect without any
liability (except for the obligations of payment arising prior to the event of
Force Majeure) towards the other Party.

14.7. Notices
Any notice given by one Party to the other shall be deemed properly given and
deeded received if specifically acknowledged by the receiving Party in writing
or when successfully delivered to the recipient by hand, fax, or special
courier during normal business hours on a business day to the addresses
specified for each Party on the signature page. Each communication and document
made or delivered by one Party to the other Party pursuant to this Agreement
shall be in the English language.

14.8. Export Control
Licensee acknowledges that the Redistributables, as incorporated in
Applications or Devices, may be subject to export control restrictions under
the applicable laws of respective countries. Licensee shall fully comply with
all applicable export license restrictions and requirements as well as with all
laws and regulations relating to the Redistributables and exercise of licenses
hereunder and shall procure all necessary governmental authorizations,
including without limitation, all necessary licenses, approvals, permissions or
consents, where necessary for the re-exportation of the Redistributables,
Applications and/or Devices.

14.9. No Implied License
There are no implied licenses or other implied rights granted under this
Agreement, and all rights, save for those expressly granted hereunder, shall
remain with The Qt Company and its licensors. In addition, no licenses or
immunities are granted to the combination of the Licensed Software with any
other software or hardware not delivered by The Qt Company under this Agreement.

14.10. Attorney Fees
The prevailing Party in any action to enforce this Agreement shall be entitled
to recover its attorney's fees and costs in connection with such action.

14.11. Privacy
Licensee acknowledges and agrees that for the purpose of this Agreement, The Qt
Company may collect, use, transfer and disclose personal data pertaining to
Designated Users as well as any other employees and directors of the Licensee
and its Contractors relevant for carrying out the intent of this Agreement. Such personal data may be collected from the Licensee or directly from the relevant individuals. The Parties acknowledge that with regard to such personal data processed hereunder, The Qt Company shall be regarded as the Data Controller under the applicable Data Protection Legislation. The Qt Company shall process any such personal data in accordance with its privacy policies and practices, which will comply with all applicable requirements of the Data Protection Legislation.

14.12. Severability
If any provision of this Agreement shall be adjudged by any court of competent jurisdiction to be unenforceable or invalid, that provision shall be limited or eliminated to the minimum extent necessary so that this Agreement shall otherwise remain in full force and effect and enforceable.

APPENDICES
The Agreement includes Appendix 1, and possibly one or more of the appendices 3-5, depending on the product(s) purchased by the Licensee, what is stated in the quote or invoice, and/or what is stated in the Licensee's License Certificate.

APPENDIX 1
The modules and/or tools that are included in the respective product - Qt for Application Development (QtAD), Qt for Device Creation (QtDC), Qt for MCUs (QtMCU), Qt 3D Studio (Qt3DS) and Qt Design Studio (QtDS) - are marked with 'X' in the below table.

Parts of the product that are permitted for distribution in object-code form only ("Redistributables") are marked with 'R' in the below table.

<table>
<thead>
<tr>
<th>Modules/Tools</th>
<th>QtAD</th>
<th>QtDC</th>
<th>QtMCU</th>
<th>Qt3DS</th>
<th>QtDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qt Core</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt GUI</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Multimedia</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Multimedia Widgets</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Network</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt QML</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick</td>
<td>X,R</td>
<td>X,R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Controls 2</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Dialogs</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Layouts</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Test</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt SQL</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Test</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Widgets</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Active Qt</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt 3D</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Android Extras</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Bluetooth</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Canvas 3D</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Concurrent</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt D-Bus</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Gamepad</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Graphical Effects</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Help</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Image Formats</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Location</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Mac Extras</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Network Authorization</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt NFC</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Platform Headers</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>Qt Positioning</td>
<td>X</td>
<td>R</td>
<td>X</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Qt Print Support</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>Qt Purchasing</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt for Python</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Controls</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Extras</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Quick Widgets</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt SCXML</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Sensors</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Serial Bus</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Serial Port</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Speech</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt SVG</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt UI Tools</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt WebChannel</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt WebEngine</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt WebSockets</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt WebView</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Windows Extras</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt X11 Extras</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt XML</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt XML Patterns</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Wayland Compositor</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Charts</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td>Qt Data Visualization</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td>X.R</td>
<td></td>
</tr>
<tr>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>Qt Virtual Keyboard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boot 2 Qt stack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt OTA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device Utilities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Debugging Bridge (QDB) Daemon</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Ultralite Controls</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Ultralite</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Creator</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Designer (Qt Widget Designer)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Quick Designer (Qt Creator plugin)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Linguist</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Assistant</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>lupdate</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>lrelease</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>qmake</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uic</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rcc</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>qlalr</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>qdoc</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>qmlscene</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>qmlviewer</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target toolchains</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qt Debugging Bridge (QDB) Host Tools</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>qtconfig-gui</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX 3: ADDITIONS TO LICENSED SOFTWARE

In addition to what is provided under the definition of the Licensed Software, Parties agree that Licensed Software shall also include the Add-On Products of The Qt Company, as mentioned in this Appendix, if included in the quote / invoice.

The Modules and/or Tools of the Licensed Software that are included with each Add-On Product respectively are marked with ‘X’ in the below table. Parts of the respective Add-On Product that are permitted for distribution in object-code form only ("Redistributables") are marked with ‘R’ in the below table.

| Add-On Product(s) | | | | | |
|-------------------|-------------------|
| Modules / Tools of Licensed Software | Qt for Automation | Qt Automotive | Qt for Renderer | Qt Application | Qt GammaRay | Qt Deployment |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |
|                   |                  |                |                 |               |             |               |

**Qt MQTT** | X,R | | | | |

**Qt KNX** | X,R | | | | |

**Qt OPC UA** | X,R | | | | |

**Qt CoAP** | X,R | | | | |

**Qt Safe** | | X,R | | | | |

**Renderer** | | | | | |
| Qt          |      |      |      |      |      |
| Application |      | X,R  |      | X,R  |      |
| Manager     |      |      |      |      |      |
| Qt IVI      |      | X,R  |      |      |      |
| Reference UI|      | X,R  |      |      |      |
| Qt GENIVI   |      | X,R  |      |      |      |
| Extras      |      |      |      |      |      |
| QML Live    |      | X    |      |      |      |
| Qt Creator  |      | X    |      |      |      |
| Deployment  |      |      |      |      |      |
| Qt Creator  |      |      |      |      |      |
| Plugin for  |      |      |      |      |      |
| Qt          |      |      |      |      |      |
| Automotive  |      |      |      |      |      |
| Suite       |      | X    |      |      |      |
| Deployment  |      |      |      |      |      |
| Server      |      |      |      |      |      |
| Qt Design   |      | X    |      |      |      |
| Studio      |      |      |      |      |      |
| Qt 3D Studio|      | X    |      |      |      |
| GammaRay    |      | X    |      |      | X    |
| Platform    |      |      |      |      |      |
| adaptations |      |      |      |      |      |
| for         |      |      |      |      | X    |
| specified   |      |      |      |      |      |
| Deployment  |      |      |      |      |      |
| Platforms   |      |      |      |      |      |
| Qt for      |      |      |      |      |      |
| Device      |      | X    |      |      |      |
| Creation    |      |      |      |      |      |

All the above Redistributables are subject to applicable provisions and
limitations including but not limited to what is defined in section 3 of the Agreement.

APPENDIX 4: SMALL BUSINESS AND STARTUP

The provisions of this Appendix 4 are applicable for Start-up Companies and for the Evaluation Term. For the purpose of this Appendix 4, the following additional definitions shall be applicable:

"Trial Term" shall mean a period of twelve (12) months.

"Start-up Company" means a company with a maximum annual revenue, including funding, equivalent to 100,000 USD (in applicable currency) during a respective calendar year, as evidenced by duly audited records of the Licensee and approved by The Qt Company.

During the Trial Term, Section 3 shall apply with following modifications ("Trial Term Modifications"):

(i) Licenses granted under Sections 3.1 and 3.2 shall be free of any charge. For clarity, License for distribution of Devices pursuant to Section 3.3 is subject to applicable License Fee for necessary Distribution Licenses;

(ii) Development License under Section 3.1 is limited to a maximum of three (3) Designated Users; and

(iii) Support is available subject to availability, as judged by The Qt Company at its free and absolute discretion, provided that support will be limited to a maximum of ten (10) tickets during the Trial Term.

Upon expiry of the Trial Term:

(a) This Appendix 4 is terminated, Trial Term Modifications cease to remain in force, Licensee's Development Licenses shall be automatically converted into licenses subject to a License Fee (in the amount specified in the quote or in Appendix 2 and payable with a 30-day payment term) and Licensee's rights and obligations under this Agreement shall continue to remain in force under the standard provisions of the Agreement, unless the Licensee notifies The Qt Company in writing no less than ninety (90) days before such expiry date that Licensee does not agree to such continuance, in which event the Agreement, and all rights of the Licensee thereunder, shall expire; provided however that

(b) in the event the Licensee still qualifies as a Start-up Company, the Licensee has an option ("Option"), instead of what is stated in item a) above, to renew the Trial Term. Renewal is limited to one time, and the total duration of Trial Term is thus 24 months after the effective date. Licensee shall notify The Qt Company in writing, no less than ninety (90) days before the expiry date, if Licensee wish to exercise the Option.
APPENDIX 5: NON-COMMERCIAL USE
The provisions of this Appendix 5 are applicable for non-commercial use of the Licensed Software by the Licensee.

For the purpose of this Appendix 5, the following additional definitions (replacing the relevant definition of the Agreement, where applicable) shall be applicable:

"Demo Units" shall mean (i) hardware development platform, which incorporates the Licensed Software along with Licensee's software and/or hardware, and (ii) prototype versions of Applications or Devices.

"Designated User(s)" shall mean the employees and students of the Licensee.

"Licensee Products" shall mean Applications and/or Devices.

"Permitted Purpose" shall mean (i) Licensee's internal evaluation and testing of Licensed Software, (ii) building Demo Units as well as (iii) educational use.

"Term" shall mean a period of twelve (12) months or any such other period as may be agreed between the Parties.

For the purpose of this Appendix 5, the following changes shall be agreed with respect to relevant Sections of the Agreement:

I. Recital (A) shall be replaced in its entirety to read as follows:

"(A) Licensee wishes to use the Licensed Software for the Permitted Purpose."

II. Section 3.1 shall be replaced in its entirety to read as follows:

"The Qt Company grants to Licensee a personal, non-exclusive, non-transferable, revocable, royalty-free license, valid for the Term, to use, modify and copy the Licensed Software solely for the Permitted Purpose.

Licensee may install copies of the Licensed Software on an unlimited number of computers provided that only Designated Users may use the Licensed Software.

Licensee may demonstrate the Demo Units, provided that such demonstrations must be conducted by Licensee, and the Demo Units must remain in Licensee's possession and under Licensee's control at all times.
For clarity, this Agreement does not (i) entitle Licensee to use Licensed Software to create Applications or Devices (other than prototypes thereof) or (ii) carry any distribution rights to Licensee, but such rights are subject to and conditional upon conclusion of a separate license agreement with The Qt Company."

III. Sections 3.2, 3.3, 8 and 10 shall be deleted.

IV. Section 3.4 shall be replaced in its entirety to read as follows:

"Licensee shall not:
- remove or alter any copyright, trademark or other proprietary rights notice contained in any portion of the Licensed Software;
- transfer, publish, sublicense, disclose, display or otherwise make the Licensed Software available to any third party (except that Licensee may demonstrate the Demo Units pursuant to Section 3.1);
- in any way combine, incorporate or integrate Licensed Software with, or use Licensed Software for creation of, any software created with or incorporating Open Source Qt;

Licensee shall cause all Designated Users who make use of the licenses granted under this Agreement, to be contractually bound to comply with the relevant terms of this Agreement and not to use the Licensed Software beyond the terms hereof. Licensee shall be responsible for any and all actions and omissions of its Designated Users relating to the Licensed Software and use thereof.

Any use of Licensed Software beyond the provisions of this Agreement is strictly prohibited and requires an additional license from The Qt Company."

V. Section 12 shall be replaced in its entirety to read as follows:

"This Agreement shall enter into force upon due acceptance by both Parties and remain in force for the Term, unless and until terminated pursuant to the terms of Section 12.

Upon termination of the Agreement, Licensee shall cease using the Licensed Software. All other copies of Licensed Software in the possession or control of Licensee must be erased or destroyed. An officer of Licensee must, upon request, promptly deliver to The Qt Company a written confirmation that this has occurred."

Except for the modifications specified above, this Appendix carries no change
to the terms of the Agreement which shall remain in full force.

// Copyright (c) 2010 The Chromium Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:

// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets """" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same """"printed page"""" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2015-2018 Google, Inc., Netflix, Inc., Microsoft Corp. and contributors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright 2005 Patrick Lam
Copyright 2009 Roozbeh Pournader
Copyright 2008,2009 Red Hat, Inc.
Copyright 2008 Danilo egan
Copyright 2012 Google, Inc.

Permission to use, copy, modify, distribute, and sell this software and its
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of usrsctp nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Roy Riojas

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2011 Einar Otto Stangvik <einaros@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
courage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating
system.

Although the Lesser General Public License is Less protective of the
users' freedom, it does ensure that the user of a program that is
linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2014 Forbes Lindesay

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

^ expected errors | v input

// Copyright (c) 2017 Mike Pennisi. All rights reserved.

// Use of this source code is governed by a BSD-style license that can be
// found in the LICENSE file.
void 0;
The JsonCpp library's source code, including accompanying documentation,
tests and demonstration applications, are licensed under the following
conditions...

The author (Baptiste Lepilleur) explicitly disclaims copyright in all
jurisdictions which recognize such a disclaimer. In such jurisdictions,
this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of
2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur, and is
released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this
software may choose to accept it either as 1) Public Domain, 2) under the
conditions of the MIT License (see below), or 3) under the terms of dual
Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is
described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

========================================================================
Copyright (c) 2007-2010 Baptiste Lepilleur

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the Public Domain "license" you can re-license your copy using whatever license you like.

Copyright (C) 2009 Apple Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

UglifyJS is released under the BSD license:

Copyright 2012-2019 (c) Mihai Bazon <mihai.bazon@gmail.com>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials
provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER AS IS AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The FreeType Project LICENSE
----------------------------

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction
============

The FreeType Project is distributed in several archive packages;
some of them may contain, in addition to the FreeType font engine,
various tools and contributions which rely on, or relate to, the
FreeType Project.

This license applies to all files found in such packages, and
which do not fall under their own explicit license. The license
affects thus the FreeType font engine, the test programs,
documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG
(Independent JPEG Group) licenses, which all encourage inclusion
and use of free software in commercial and freeware products
alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be
  interested in any kind of bug reports. ('as is' distribution)

- You can use this software for whatever you want, in parts or
full form, without having to pay us. (‘royalty-free’ usage)

- You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. (‘credits’)

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```---
Portions of this software are copyright <year> The FreeType Project (www.freetype.org). All rights reserved.
---
```

Please replace `<year>` with the value from the FreeType version you actually use.

Legal Terms
============

0. Definitions
-------------

Throughout this license, the terms `package`, `FreeType Project`, and `FreeType archive` refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project`, be they named as alpha, beta or final release.

`You` refers to the licensee, or person using the project, where `using` is a generic term including compiling the project’s source code as well as linking it to form a `program` or `executable`. This program is referred to as `a program using the FreeType engine`.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive.
If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

--------------

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

--------------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file ("FTL.TXT") unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

--------------
Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: 'FreeType Project', 'FreeType Engine', 'FreeType library', or 'FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

 o freetype@nongnu.org

 Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

 o freetype-devel@nongnu.org

 Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

 Our home page can be found at

 http://www.freetype.org

--- end of FTL.TXT ---

MIT License

Copyright (c) 2012 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Permission to use, copy, modify, distribute, and sell this
software and its documentation for any purpose is hereby granted
without fee, provided that the above copyright notice appear in
all copies and that both that copyright notice and this permission
notice appear in supporting documentation, and that the name of
the copyright holders not be used in advertising or publicity
pertaining to distribution of the software without specific,
written prior permission. The copyright holders make no
representations about the suitability of this software for any
purpose. It is provided "as is" without express or implied
warranty.

THE COPYRIGHT HOLDERS DISCLAIM ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN
AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

GNU Free Documentation License
Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other
functional and useful document "free" in the sense of freedom: to
assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles
are designated, as being those of Invariant Sections, in the notice 
that says that the Document is released under this License. If a 
section does not fit the above definition of Secondary then it is not 
allowed to be designated as Invariant. The Document may contain zero 
Invariant Sections. If the Document does not identify any Invariant 
Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, 
as Front-Cover Texts or Back-Cover Texts, in the notice that says that 
the Document is released under this License. A Front-Cover Text may 
be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, 
represented in a format whose specification is available to the 
general public, that is suitable for revising the document 
straightforwardly with generic text editors or (for images composed of 
pixels) generic paint programs or (for drawings) some widely available 
drawing editor, and that is suitable for input to text formatters or 
for automatic translation to a variety of formats suitable for input 
to text formatters. A copy made in an otherwise Transparent file 
format whose markup, or absence of markup, has been arranged to thwart 
or discourage subsequent modification by readers is not Transparent. 
An image format is not Transparent if used for any substantial amount 
of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain 
ASCII without markup, Texinfo input format, LaTeX input format, SGML 
or XML using a publicly available DTD, and standard-conforming simple 
HTML, PostScript or PDF designed for human modification. Examples of 
transparent image formats include PNG, XCF and JPG. Opaque formats 
include proprietary formats that can be read and edited only by 
proprietary word processors, SGML or XML for which the DTD and/or 
processing tools are not generally available, and the 
machine-generated HTML, PostScript or PDF produced by some word 
processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, 
plus such following pages as are needed to hold, legibly, the material 
this License requires to appear in the title page. For works in 
formats which do not have any title page as such, "Title Page" means 
the text near the most prominent appearance of the work's title, 
preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose 
title either is precisely XYZ or contains XYZ in parentheses following 
text that translates XYZ in another language. (Here XYZ stands for a 
specific section name mentioned below, such as "Acknowledgements", 
"Dedications", "Endorsements", or "History"). To "Preserve the Title"
of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties; any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering
more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material.

If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add
to it an item stating at least the title, year, new authors, and
publisher of the Modified Version as given on the Title Page. If
there is no section Entitled "History" in the Document, create one
stating the title, year, authors, and publisher of the Document as
given on its Title Page, then add an item describing the Modified
Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for
public access to a Transparent copy of the Document, and likewise
the network locations given in the Document for previous versions
it was based on. These may be placed in the "History" section.
You may omit a network location for a work that was published at
least four years before the Document itself, or if the original
publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications",
Preserve the Title of the section, and preserve in the section all
the substance and tone of each of the contributor acknowledgements
and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document,
unaltered in their text and in their titles. Section numbers
or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section
may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements"
or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or
appendices that qualify as Secondary Sections and contain no material
copied from the Document, you may at your option designate some or all
of these sections as invariant. To do this, add their titles to the
list of Invariant Sections in the Modified Version's license notice.
These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains
nothing but endorsements of your Modified Version by various
parties—for example, statements of peer review or that the text has
been approved by an organization as the authoritative definition of a
standard.

You may add a passage of up to five words as a Front-Cover Text, and a
passage of up to 25 words as a Back-Cover Text, to the end of the list
of Cover Texts in the Modified Version. Only one passage of
Front-Cover Text and one of Back-Cover Text may be added by (or
through arrangements made by) any one entity. If the Document already
includes a cover text for the same cover, previously added by you or
by arrangement made by the same entity you are acting on behalf of,
you may not add another; but you may replace the old one, on explicit
permission from the previous publisher that added the old one.
The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate
and independent documents or works, in or on a volume of a storage or
distribution medium, is called an "aggregate" if the copyright
resulting from the compilation is not used to limit the legal rights
of the compilation's users beyond what the individual works permit.
When the Document is included in an aggregate, this License does not
apply to the other works in the aggregate which are not themselves
derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these
copies of the Document, then if the Document is less than one half of
the entire aggregate, the Document's Cover Texts may be placed on
covers that bracket the Document within the aggregate, or the
electronic equivalent of covers if the Document is in electronic form.
Otherwise they must appear on printed covers that bracket the whole
aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may
distribute translations of the Document under the terms of section 4.
Replacing Invariant Sections with translations requires special
permission from their copyright holders, but you may include
translations of some or all Invariant Sections in addition to the
original versions of these Invariant Sections. You may include a
translation of this License, and all the license notices in the
Document, and any Warranty Disclaimers, provided that you also include
the original English version of this License and the original versions
of those notices and disclaimers. In case of a disagreement between
the translation and the original version of this License or a notice
or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements",
"Dedications", or "History", the requirement (section 4) to Preserve
its Title (section 1) will typically require changing the actual
title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except
as expressly provided for under this License. Any other attempt to
copy, modify, sublicense or distribute the Document is void, and will
automatically terminate your rights under this License. However,
parties who have received copies, or rights, from you under this
License will not have their licenses terminated so long as such
parties remain in full compliance.
10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”.

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.
Copyright (c) 2010 Jonathan Hartley
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors
  may be used to endorse or promote products derived from this software without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

From __init__.py:

# COPYRIGHT AND LICENSE
#
# Copyright 2001 Gareth Rees.  All rights reserved.
# Copyright 2004-2013 Ned Batchelder. All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are
# met:
#
# # 1. Redistributions of source code must retain the above copyright
#    notice, this list of conditions and the following disclaimer.
# #
# # 2. Redistributions in binary form must reproduce the above copyright
#    notice, this list of conditions and the following disclaimer in the
#    documentation and/or other materials provided with the
#    distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

Copyright 2018-2019, VideoLAN and dav1d authors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014 Component contributors <dev@component.io>

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This package contains code originally written by Isaac Z. Schlueter.
Used with permission.

Copyright (c) Meryn Stol ("Author")
All rights reserved.

The BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014-2018 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Here, glslang proper means core GLSL parsing, HLSL parsing, and SPIR-V code generation. Glslang proper requires use of two licenses, one that covers non-preprocessing and an additional one that covers preprocessing.

Bison was removed long ago. You can build glslang from the source grammar, using tools of your choice, without using bison or any bison files.

Other parts, outside of glslang proper, include:

- gl_types.h, only needed for OpenGL-like reflection, and can be left out of a parse and codegen project. See it for its license.

- update_glslang_sources.py, which is not part of the project proper and does not need to be used.

- the SPIR-V "remapper", which is optional, but has the same license as glslang proper

- Google tests and SPIR-V tools, and anything in the external subdirectory are external and optional; see them for their respective licenses.

The core of glslang-proper, minus the preprocessor is licenced as follows:

//
// Copyright (C) 2015-2018 Google, Inc.
// Copyright (C) <various other dates and companies>
//
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
//
// Redistributions of source code must retain the above copyright
The preprocessor has the core license stated above, plus an additional licence:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Corporation may be used to endorse or promote products derived from the NVIDIA Software without specific prior written permission from NVIDIA. Except as expressly stated in this notice, no other rights or licenses express or implied, are granted by NVIDIA herein, including but not limited to any patent rights that may be infringed by your derivative works or by other works in which the NVIDIA Software may be incorporated. No hardware is licensed hereunder.

THE NVIDIA SOFTWARE IS BEING PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR ITS USE AND OPERATION EITHER ALONE OR IN COMBINATION WITH OTHER PRODUCTS.

IN NO EVENT SHALL NVIDIA BE LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL, EXEMPLARY, CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, LOST PROFITS; PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) OR ARISING IN ANY WAY OUT OF THE USE, REPRODUCTION, MODIFICATION AND/OR DISTRIBUTION OF THE NVIDIA SOFTWARE, HOWEVER CAUSED AND WHETHER UNDER THEORY OF CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY OR OTHERWISE, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Gal Koren
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (C) 2014 Rase-

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2017-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/*
 *
 * Copyright (c) 2001-2017 Cisco Systems, Inc.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * Redistributions in binary form must reproduce the above
 * copyright notice, this list of conditions and the following
 * disclaimer in the documentation and/or other materials provided
 * with the distribution.
 *
 * Neither the name of the Cisco Systems, Inc. nor the names of its
 * contributors may be used to endorse or promote products derived
 * from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
 * INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 *
*/

Poly2Tri Copyright (c) 2009-2010, Poly2Tri Contributors
http://code.google.com/p/poly2tri/

All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Poly2Tri nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2002-2006 Marcus Geelnard
Copyright (c) 2006-2016 Camilla Lwy <elmindreda@glfw.org>

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

Copyright (c) 2016, Mapbox

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above
* You may use, copy, modify and distribute this code for any purpose (include
* commercial use) and without fee. Please refer to this package when you modify
* this code.
*/

Copyright (C) 2016 Intel Corporation.
Copyright (C) 2015 Klarlvdalens Datakonsult AB, a KDAB Group company, info@kdab.com

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2014 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
Parts of the following directories are available under Apache v2.0

src/de
Copyright (c) 2009-2011 Christian Kohlschutter

third_party/gwt_exporter
Copyright 2007 Timepedia.org

third_party/gwt-2.5.1
Copyright 2008 Google

java/org/chromium/distiller/dev
Copyright 2008 Google

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
The MIT License (MIT)

Copyright (c) 2015 JD Ballard
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2012
MIPS Technologies, Inc., California.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the MIPS Technologies, Inc., nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE MIPS TECHNOLOGIES, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE MIPS TECHNOLOGIES, INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Authors:
Branimir Vasic   (bvasic@mips.com)
Darko Laus       (darko@mips.com)
MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2013 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// software.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
The MIT License (MIT)

Copyright (c) 2014-2017 Calvin Metcalf, Fedor Indutny & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# spdx-license-ids


A list of [SPDX license](https://spdx.org/licenses/) identifiers

## Installation

[Download JSON directly](https://raw.githubusercontent.com/shinnn/spdx-license-ids/master/index.json), or [use](https://docs.npmjs.com/cli/install) [npm](https://docs.npmjs.com/about-npm):

```bash
npm install spdx-license-ids
```

## [Node.js](https://nodejs/) API

```javascript
require('spdx-license-ids')
```
Type: `string[]`

All license IDs except for the currently deprecated ones.

```javascript
const ids = require('spdx-license-ids');
//=> ['0BSD', 'AAL', 'ADSL', 'AFL-1.1', 'AFL-1.2', 'AFL-2.0', 'AFL-2.1', 'AFL-3.0', 'AGPL-1.0-only', ...]

ids.includes('BSD-3-Clause'); //=> true
ids.includes('CC-BY-1.0'); //=> true

ids.includes('GPL-3.0'); //=> false
```

### require('spdx-license-ids/deprecated')

Type: `string[]`

Deprecated license IDs.

```javascript
const deprecatedIds = require('spdx-license-ids/deprecated');
//=> ['AGPL-1.0', 'AGPL-3.0', 'GFDL-1.1', 'GFDL-1.2', 'GFDL-1.3', 'GPL-1.0', 'GPL-2.0', ...]

deprecatedIds.includes('BSD-3-Clause'); //=> false
deprecatedIds.includes('CC-BY-1.0'); //=> false

deprecatedIds.includes('GPL-3.0'); //=> true
```

## License

[Creative Commons Zero v1.0 Universal](https://creativecommons.org/publicdomain/zero/1.0/deed)

Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors
Copyright (c) 2013 Aleksey Tulinov <aleksey.tulinov@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Notice some of these images are derived from Google applications resources. They were provided under the following license:

You may use the materials in this directory without restriction to develop your apps and to use in your apps.

# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# Names should be added to this file only after verifying that
# the individual or the individual's organization has agreed to
# the appropriate Contributor License Agreement, found here:
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# The agreement for individuals can be filled out on the web.
#
# When adding J Random Contributor's name to this file,
# either J's name or J's organization's name should be
# added to the AUTHORS file, depending on whether the
# individual or corporate CLA was used.
#
# Names should be added to this file as:
#   Name <email address>
#
# Please keep the list sorted.

Eric Veach <ericv@google.com>
Jesse Rosenstock <jmr@google.com>
Julien Basch <julienbasch@google.com>

Node.js is licensed for use as follows:

====

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

This license applies to parts of Node.js originating from the https://github.com/joyent/node repository:

---

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

The Node.js license applies to all parts of Node.js that are not externally maintained libraries.

The externally maintained libraries used by Node.js are:

- c-ares, located at deps/cares, is licensed as follows:

  ---

  Copyright 1998 by the Massachusetts Institute of Technology.
  Copyright (C) 2007-2013 by Daniel Stenberg

  Permission to use, copy, modify, and distribute this
software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

- HTTP Parser, located at deps/http_parser, is licensed as follows:

  http_parser.c is based on src/http/ngx_http_parse.c from NGINX copyright Igor Sysoev.

  Additional changes are licensed under the same terms as NGINX and copyright Joyent, Inc. and other Node contributors. All rights reserved.

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- ICU, located at deps/icu-small, is licensed as follows:

  COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

  Copyright 1991-2016 Unicode, Inc. All rights reserved.
  Distributed under the Terms of Use in http://www.unicode.org/copyright.html

  Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY
SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale, use
or other dealings in this Software without prior written authorization
of the copyright holder.

All trademarks and registered trademarks mentioned herein are the
property of their respective owners.


# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
# Redistributions of source code must retain the above copyright notice,
# this list of conditions and the following disclaimer.
# Redistributions in binary form must reproduce the above
# copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided with
# the distribution.
# Neither the name of Google Inc. nor the names of its
# contributors may be used to endorse or promote products derived from
# this software without specific prior written permission.
#
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
# DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
# LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
# CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
# SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
# BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
# LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
# SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
#
# The word list in cjdict.txt are generated by combining three word lists
# listed below with further processing for compound word breaking. The
# frequency is generated with an iterative training against Google web
# corpora.
#
# * Libtabe (Chinese)
#   - https://sourceforge.net/project/?group_id=1519
#   - Its license terms and conditions are shown below.
#
# * IPADIC (Japanese)
#   - http://chasen.aist-nara.ac.jp/chasen/distribution.html
#   - Its license terms and conditions are shown below.
#
# ---------COPYING.libtabe ---- BEGIN---------------------
#
# /*
#  * Copyright(c) 1999 TaBE Project.
#  * Copyright(c) 1999 Pai-Hsiang Hsiao.
#  * All rights reserved.
#  *
#  * Redistribution and use in source and binary forms, with or without
#  * modification, are permitted provided that the following conditions
#  * are met:
#  *
#  * Redistributions of source code must retain the above copyright
#  * notice, this list of conditions and the following disclaimer.
#  * Redistributions in binary form must reproduce the above copyright
#  * notice, this list of conditions and the following disclaimer in
#  * the documentation and/or other materials provided with the
#  * distribution.
#  * Neither the name of the TaBE Project nor the names of its
#  * contributors may be used to endorse or promote products derived
#  * from this software without specific prior written permission.
#  *
#  */

Open Source Used In DNAC Platform Arica 6131
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# /*
# * Copyright (c) 1999 Computer Systems and Communication Lab,
# * Institute of Information Science, Academia
# * Sinica. All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * . Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * . Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * . Neither the name of the Computer Systems and Communication Lab
# * nor the names of its contributors may be used to endorse or
# * promote products derived from this software without specific
# * prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
University of Illinois
c-tsa4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsa4

--------------COPYING.libtabe-----END--------------------------------

--------------COPYING.ipadic-----BEGIN-------------------------------

and Technology. All Rights Reserved.

Use, reproduction, and distribution of this software is permitted.
Any copy of this software, whether in its original form or modified,
must include both the above copyright notice and the following
paragraphs.

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
IMPLIED, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
# Each user of the program will agree and understand, and be deemed to
# have agreed and understood, that there is no warranty whatsoever for
# the program and, accordingly, the entire risk arising from or
# otherwise connected with the program is assumed by the user.
#
# Therefore, neither ICOT, the copyright holder, or any other
# organization that participated in or was otherwise related to the
# development of the program and their respective officials, directors,
# officers and other employees shall be held liable for any and all
# damages, including, without limitation, general, special, incidental
# and consequential damages, arising out of or otherwise in connection
# with the use or inability to use the program or any product, material
# or result produced or otherwise obtained by using the program,
# regardless of whether they have been advised of, or otherwise had
# knowledge of, the possibility of such damages at any time during the
# project or thereafter. Each user will be deemed to have agreed to the
# foregoing by his or her commencement of use of the program. The term
# "use" as used herein includes, but is not limited to, the use,
# modification, copying and distribution of the program and the
# production of secondary products from the program.
#
# In the case where the program, whether in its original form or
# modified, was distributed or delivered to or received by a user from
# any person, organization or entity other than ICOT, unless it makes or
# grants independently of ICOT any specific warranty to the user in
# writing, such person, organization or entity, will also be exempted
# from and not be held liable to the user for any such damages as noted
# above as far as the program is concerned.
#
# ---------------COPYING.ipadic-----END----------------------------------

3. Lao Word Break Dictionary Data (laodict.txt)

# Copyright (c) 2013 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
#      (copied below)
#
# This file is derived from the above dictionary, with slight
# modifications.
# -----------------------------------------------
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification,
# are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in
# binary form must reproduce the above copyright notice, this list of
# conditions and the following disclaimer in the documentation and/or
# other materials provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
#  ----------------------------------------------------------------------

4. Burmese Word Break Dictionary Data (burmesedict.txt)

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
#   github.com/kanyawtech/myanmar-karen-word-lists
#
# ----------------------------------------------------------------------

# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
# disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided
# with the distribution.
5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

# 7. Database Ownership
#
# The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

- libuv, located at deps/uv, is licensed as follows:

  libuv is part of the Node project: http://nodejs.org/
  libuv may be distributed alone under Node's license:
Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This license applies to all parts of libuv that are not externally maintained libraries.

The externally maintained libraries used by libuv are:

- tree.h (from FreeBSD), copyright Niels Provos. Two clause BSD license.

- inet_pton and inet_ntop implementations, contained in src/inet.c, are copyright the Internet Systems Consortium, Inc., and licensed under the ISC license.

- stdint-msvc2008.h (from mstdint), copyright Alexander Chemeris. Three clause BSD license.

- pthread-fixes.h, pthread-fixes.c, copyright Google Inc. and Sony Mobile Communications AB. Three clause BSD license.


- OpenSSL, located at deps/openssl, is licensed as follows:

---
Copyright (c) 1998-2016 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

====================================================================

This product includes cryptographic software written by Eric Young
Open Source Used In DNAC Platform Arica  6139

(eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

- Punycode.js, located at lib/punycode.js, is licensed as follows:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- V8, located at deps/v8, is licensed as follows:

This license applies to all parts of V8 that are not externally maintained libraries. The externally maintained libraries used by V8 are:

- PCRE test suite, located in test/mjsunit/third_party/regexp-pcre/regexp-pcre.js. This is based on the test suite from PCRE-7.3, which is copyrighted by the University of Cambridge and Google, Inc. The copyright notice and license are embedded in regexp-pcre.js.

- Layout tests, located in test/mjsunit/third_party/object-keys. These are based on layout tests from webkit.org which are copyrighted by Apple Computer, Inc. and released under a 3-clause BSD license.

- Strongtalk assembler, the basis of the files assembler-arm-inl.h, assembler-arm.cc, assembler-arm.h, assembler-ia32-inl.h, assembler-ia32.cc, assembler-ia32.h, assembler-x64-inl.h, assembler-x64.cc, assembler-x64.h, assembler-mips-inl.h,
assembler-mips.cc, assembler-mips.h, assembler.cc and assembler.h. This code is copyrighted by Sun Microsystems Inc. and released under a 3-clause BSD license.

- Valgrind client API header, located at third_party/valgrind/valgrind.h
  This is release under the BSD license.

These libraries have their own licenses; we recommend you read them, as their terms may differ from the terms below.

Further license information can be found in LICENSE files located in sub-directories.

Copyright 2014, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

- zlib, located at deps/zlib, is licensed as follows:

  zlib.h -- interface of the 'zlib' general purpose compression library
  version 1.2.8, April 28th, 2013

  Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly       Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

---

- npm, located at deps/npm, is licensed as follows:

---

The npm application
Copyright (c) npm, Inc. and Contributors
Licensed on the terms of The Artistic License 2.0

Node package dependencies of the npm application
Copyright (c) their respective copyright owners
Licensed on their respective license terms

The npm public registry at https://registry.npmjs.org
and the npm website at https://www.npmjs.com
Operated by npm, Inc.
Use governed by terms published on https://www.npmjs.com

"Node.js"
Neither npm nor npm, Inc. are affiliated with Joyent, Inc.

The Node.js application
Project of Node Foundation, https://nodejs.org

The npm Logo
Copyright (c) Mathias Pettersson and Brian Hammond

"Gubblebum Blocky” typeface
Copyright (c) Tjarda Koster, https://jelloween.deviantart.com
Used with permission
The Artistic License 2.0

Copyright (c) 2000-2006, The Perl Foundation.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

This license establishes the terms under which a given free software Package may be copied, modified, distributed, and/or redistributed. The intent is that the Copyright Holder maintains some artistic control over the development of that Package while still keeping the Package available as open source and free software.

You are always permitted to make arrangements wholly outside of this license directly with the Copyright Holder of a given Package. If the terms of this license do not permit the full use that you propose to make of the Package, you should contact the Copyright Holder and seek a different licensing arrangement.

Definitions

"Copyright Holder" means the individual(s) or organization(s) named in the copyright notice for the entire Package.

"Contributor" means any party that has contributed code or other material to the Package, in accordance with the Copyright Holder's procedures.

"You" and "your" means any person who would like to copy, distribute, or modify the Package.

"Package" means the collection of files distributed by the Copyright Holder, and derivatives of that collection and/or of those files. A given Package may consist of either the Standard Version, or a Modified Version.

"Distribute" means providing a copy of the Package or making it accessible to anyone else, or in the case of a company or organization, to others outside of your company or organization.

"Distributor Fee" means any fee that you charge for Distributing this Package or providing support for this Package to another party. It does not mean licensing fees.
"Standard Version" refers to the Package if it has not been modified, or has been modified only in ways explicitly requested by the Copyright Holder.

"Modified Version" means the Package, if it has been changed, and such changes were not explicitly requested by the Copyright Holder.

"Original License" means this Artistic License as Distributed with the Standard Version of the Package, in its current version or as it may be modified by The Perl Foundation in the future.

"Source" form means the source code, documentation source, and configuration files for the Package.

"Compiled" form means the compiled bytecode, object code, binary, or any other form resulting from mechanical transformation or translation of the Source form.

Permission for Use and Modification Without Distribution

(1) You are permitted to use the Standard Version and create and use Modified Versions for any purpose without restriction, provided that you do not Distribute the Modified Version.

Permissions for Redistribution of the Standard Version

(2) You may Distribute verbatim copies of the Source form of the Standard Version of this Package in any medium without restriction, either gratis or for a Distributor Fee, provided that you duplicate all of the original copyright notices and associated disclaimers. At your discretion, such verbatim copies may or may not include a Compiled form of the Package.

(3) You may apply any bug fixes, portability changes, and other modifications made available from the Copyright Holder. The resulting Package will still be considered the Standard Version, and as such will be subject to the Original License.

Distribution of Modified Versions of the Package as Source

(4) You may Distribute your Modified Version as Source (either gratis or for a Distributor Fee, and with or without a Compiled form of the Modified Version) provided that you clearly document how it differs from the Standard Version, including, but not limited to, documenting any non-standard features, executables, or modules, and provided that you do at least ONE of the following:
(a) make the Modified Version available to the Copyright Holder of the Standard Version, under the Original License, so that the Copyright Holder may include your modifications in the Standard Version.

(b) ensure that installation of your Modified Version does not prevent the user installing or running the Standard Version. In addition, the Modified Version must bear a name that is different from the name of the Standard Version.

(c) allow anyone who receives a copy of the Modified Version to make the Source form of the Modified Version available to others under

(i) the Original License or

(ii) a license that permits the licensee to freely copy, modify and redistribute the Modified Version using the same licensing terms that apply to the copy that the licensee received, and requires that the Source form of the Modified Version, and of any works derived from it, be made freely available in that license fees are prohibited but Distributor Fees are allowed.

Distribution of Compiled Forms of the Standard Version or Modified Versions without the Source

(5) You may Distribute Compiled forms of the Standard Version without the Source, provided that you include complete instructions on how to get the Source of the Standard Version. Such instructions must be valid at the time of your distribution. If these instructions, at any time while you are carrying out such distribution, become invalid, you must provide new instructions on demand or cease further distribution. If you provide valid instructions or cease distribution within thirty days after you become aware that the instructions are invalid, then you do not forfeit any of your rights under this license.

(6) You may Distribute a Modified Version in Compiled form without the Source, provided that you comply with Section 4 with respect to the Source of the Modified Version.

Aggregating or Linking the Package

(7) You may aggregate the Package (either the Standard Version or Modified Version) with other packages and Distribute the resulting aggregation provided that you do not charge a licensing fee for the Package. Distributor Fees are permitted, and licensing fees for other
components in the aggregation are permitted. The terms of this license apply to the use and Distribution of the Standard or Modified Versions as included in the aggregation.

(8) You are permitted to link Modified and Standard Versions with other works, to embed the Package in a larger work of your own, or to build stand-alone binary or bytecode versions of applications that include the Package, and Distribute the result without restriction, provided the result does not expose a direct interface to the Package.

Items That are Not Considered Part of a Modified Version

(9) Works (including, but not limited to, modules and scripts) that merely extend or make use of the Package, do not, by themselves, cause the Package to be a Modified Version. In addition, such works are not considered parts of the Package itself, and are not subject to the terms of this license.

General Provisions

(10) Any use, modification, and distribution of the Standard or Modified Versions is governed by this Artistic License. By using, modifying or distributing the Package, you accept this license. Do not use, modify, or distribute the Package, if you do not accept this license.

(11) If your Modified Version has been derived from a Modified Version made by someone other than you, you are nevertheless required to ensure that your Modified Version complies with the requirements of this license.

(12) This license does not grant you the right to use any trademark, service mark, tradename, or logo of the Copyright Holder.

(13) This license includes the non-exclusive, worldwide, free-of-charge patent license to make, have made, use, offer to sell, sell, import and otherwise transfer the Package with respect to any patent claims licensable by the Copyright Holder that are necessarily infringed by the Package. If you institute patent litigation (including a cross-claim or counterclaim) against any party alleging that the Package constitutes direct or contributory patent infringement, then this Artistic License to you shall terminate on the date that such litigation is filed.

(14) Disclaimer of Warranty:
THE PACKAGE IS PROVIDED BY THE COPYRIGHT HOLDER AND CONTRIBUTORS "AS IS' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES. THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR
NON-INFRINGEMENT ARE DISCLAIMED TO THE EXTENT PERMITTED BY YOUR LOCAL LAW. UNLESS REQUIRED BY LAW, NO COPYRIGHT HOLDER OR CONTRIBUTOR WILL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING IN ANY WAY OUT OF THE USE OF THE PACKAGE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------

- GYP, located at tools/gyp, is licensed as follows:

Copyright (c) 2009 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------

- marked, located at tools/doc/node_modules/marked, is licensed as follows:

Copyright (c) 2011-2014, Christopher Jeffrey (https://github.com/chjj/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- cpplint.py, located at tools/cpplint.py, is licensed as follows:

Copyright (c) 2009 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- ESLint, located at tools/eslint, is licensed as follows:
ESLint
Copyright jQuery Foundation and other contributors, https://jquery.org/

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

---

- gtest, located at deps/gtest, is licensed as follows:

---

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- node-weak, located at test/gc/node_modules/weak, is licensed as follows:

Copyright (c) 2011, Ben Noordhuis <info@bnoordhuis.nl>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

- v8_inspector, located at deps/v8_inspector/third_party/v8_inspector, is licensed as follows:

// Copyright 2015 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
- jinja2, located at deps/v8_inspector/third_party/jinja2, is licensed as follows:

Copyright (c) 2009 by the Jinja Team, see AUTHORS for more details.

Some rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- markupsafe, located at deps/v8_inspector/third_party/markupsafe, is licensed as follows:

Copyright (c) 2010 by Armin Ronacher and contributors. See AUTHORS for more details.

Some rights reserved.
Redistribution and use in source and binary forms of the software as well as documentation, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE AND DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License, Version 2.0

Copyright (c) 2011 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and
its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This is a copy of CONTRIBUTORS file for the Pthreads-win32 library, downloaded from http://sourceware.org/cgi-bin/cvsweb.cgi/~checkout~/pthreads/CONTRIBUTORS?rev=1.32&cvsroot=pthreads-win32

Included here to compliment the Pthreads-win32 license header in wtf/ThreadingWin.cpp file. WebKit is using derived sources of ThreadCondition code from Pthreads-win32.

Contributors (in approximate order of appearance)
[See also the ChangeLog file where individuals are attributed in log entries. Likewise in the FAQ file.]

Ben Elliston
bje at cygnus dot com
Initiated the project;
setup the project infrastructure (CVS, web page, etc.);
early prototype routines.
Ross Johnson
rpj at callisto dot canberra dot edu dot au
early prototype routines;
ongoing project coordination/maintenance;
implementation of spin locks and barriers;
various enhancements;
bug fixes;
documentation;
testsuite.
Robert Colquhoun
rjc at trump dot net dot au
Early bug fixes.
John E. Bossom
John dot Bossom at cognos dot com
Contributed substantial original working implementation;
bug fixes;
ongoing guidance and standards interpretation.
Anders Norlander
anorland at hem2 dot passagen dot se
Early enhancements and runtime checking for supported Win32 routines.
Tor Lillqvist
tml at iki dot fi
General enhancements;
early bug fixes to condition variables.
Scott Lightner
scott at curriculum dot com
Bug fix.
Kevin Ruland
Kevin dot Ruland at anheuser-busch dot com
Various bug fixes.
Mike Russo
miker at eai dot com
Bug fix.
Mark E. Armstrong
avail at pacbell dot net
Bug fixes.
Lorin Hochstein
lmh at xiphos dot ca
general bug fixes; bug fixes to condition variables.
Peter Slacik
Peter dot Slacik at tatramed dot sk
Bug fixes.
Mumit Khan
khan at xraylith dot wisc dot edu
Fixes to work with Mingw32.
Milan Gardian
mg at tatramed dot sk
Bug fixes and reports/analyses of obscure problems.
Aurelio Medina
aureliom at.crt dot.com
First implementation of read-write locks.
Graham Dumpleton
Graham dot Dumpleton at ra dot pad dot otc dot telstra dot com dot au
Bug fix in condition variables.
Tristan Savatier at mpegtv dot com  
WinCE port.

Erik Hensema at hensema xs4all dot nl  
Bug fixes.

Rich Peters at micro-magic dot com

Todd Owen at lucidcalm dot dropbear dot id dot au  
Bug fixes to dll loading.

Jason Nye at nbn dot nb dot ca  
Implementation of async cancelation.

Fred Forester at eticomm dot net

Kevin D. Clark at cabletron dot com

David Baggett at itasoftware dot com  
Bug fixes.

Paul Redondo at matchvision dot com

Scott McCaskill at 3dfx dot com  
Bug fixes.

Jef Gearhart at tpssys dot com  
Bug fix.

Arthur Kantor at bexusa dot com

Steven Reddies at esemer dot com dot au  
Bug fix.

Alexander Terekhov at de dot ibm dot com  
Re-implemented and improved read-write locks;
(with Louis Thomas) re-implemented and improved
condition variables;
enhancements to semaphores;
enhancements to mutexes;
new mutex implementation in 'futex' style;
suggested a robust implementation of pthread_once
similar to that implemented by V.Kliathcko;
system clock change handling re CV timeouts;
bug fixes.

Thomas Pfaff at gmx dot net

Changes to make C version usable with C++ applications;
re-implemented mutex routines to avoid Win32 mutexes
and TryEnterCriticalSection;
procedure to fix Mingw32 thread-safety issues.

Franco Bez at gmxdotbez dot gmxdotde

Louis Thomas at arbitrade dot com
(with Alexander Terekhov) re-implemented and improved
condition variables.

David Korn at research dot att dot com
Ported to UWIN.

Phil Frisbie, Jr at hawksoft dot com

Bug fix.

Ralf Brese at pdb4 dot siemens dot de
Bug fix.
prionx at juno dot com prionx at juno dot com
Bug fixes.
Max Woodburyamtew at cds dot duke dot edu
POSIX versioning conditionals;
reduced namespace pollution;
idea to separate routines to reduce statically
linked image sizes.
Rob Fannerrfanner at stonethree dot com
Bug fix.
Michael Johnson michaelj at maine dot rr dot com
Bug fix.
Nicolas Barryboozai at yahoo dot com
Bug fixes.
Piet van Bruggenpietvb at newbridges dot nl
Bug fix.
Makoto Katoraven at oldskool dot jp
AMD64 port.
Panagiotis E. Hadjidoukashe at hpclab dot ceid dot upatras dot gr
Contributed the QueueUserAPCEX package which
makes preemptive async cancelation possible.
Will Bryantwill dot bryant at ecosm dot com
Borland compiler patch and makefile.
Anuj Goyalanuj dot goyal at gmail dot com
Port to Digital Mars compiler.
Gottlob Fregegottlobfrege at gmail dot com
re-implemented pthread_once (version 2)
(pthread_once cancellation added by rpj).
Vladimir Kliatchkovladimir at kliatchko dot com
reimplemented pthread_once with the same form
as described by A.Terekhov (later version 2):
implementation of MCS (Mellor-Crummey/Scott) locks.
A. HISTORY OF THE SOFTWARE
==================================

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations, which became
Zope Corporation. In 2001, the Python Software Foundation (PSF, see https://www.python.org/psf/) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation was a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived from</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>1.3 thru 1.5.2 1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2 and above</td>
<td>2.1.1</td>
<td>2001-now</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

=================================================================================

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

=================================================================================

Open Source Used In DNAC Platform Arica 6156
1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.


3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.
CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2
--------------------------------------------------

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

mapbox-gl-native copyright (c) 2014-2017 Mapbox.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===========================================================================

Mapbox GL uses portions of Android Gesture Detectors Framework.

Copyright (c) 2012, Almer Thie

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
Mapbox GL uses portions of Android Support Library.

Copyright (c) 2005-2013, The Android Open Source Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Mapbox GL uses portions of Boost.

Distributed under the Boost Software License, Version 1.0.

http://www.boost.org/LICENSE_1_0.txt

Mapbox GL uses portions of Clipper.

Author : Angus Johnson
Version : 6.1.3a
Date : 22 January 2014
Website : http://www.angusj.com
Copyright : Angus Johnson 2010-2014

License:
Use, modification & distribution is subject to Boost Software License Ver 1.
http://www.boost.org/LICENSE_1_0.txt

Attributions:
The code in this library is an extension of Bala Vatti's clipping algorithm:
"A generic solution to polygon clipping"
http://portal.acm.org/citation.cfm?id=129906
Computer graphics and geometric modeling: implementation and algorithms
By Max K. Agoston
Springer; 1 edition (January 4, 2005)
http://books.google.com/books?q=vatti+clipping+agoston

See also:
"Polygon Offsetting by Computing Winding Numbers"
Paper no. DETC2005-85513 pp. 565-575
ASME 2005 International Design Engineering Technical Conferences
and Computers and Information in Engineering Conference (IDETC/CIE2005)
September 24-28, 2005, Long Beach, California, USA
http://www.me.berkeley.edu/~mcmains/pubs/DAC05OffsetPolygon.pdf

===========================================================================

Mapbox GL uses portions of BugshotKit.

The MIT License (MIT)

Copyright (c) 2014 marcoarment

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

===========================================================================

Mapbox GL uses portions of CSS Color Parser.

(c) Dean McNamee <dean@gmail.com>, 2012.
C++ port by Konstantin Kfer <mail@kkaefer.com>, 2014.

https://github.com/deanm/css-color-parser-js
https://github.com/kkaefer/css-color-parser-cpp
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

===========================================================================

Mapbox GL uses portions of GLFW.

Copyright (c) 2002-2006 Marcus Geelnard
Copyright (c) 2006-2010 Camilla Berglund <elmindreda@elmindreda.org>

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

===========================================================================

Mapbox GL uses portions of libc++.
The libc++ library is dual licensed under both the University of Illinois "BSD-Like" license and the MIT license. As a user of this code you may choose to use it under either license. As a contributor, you agree to allow your code to be used under both.

Full text of the relevant licenses is included below.

====

University of Illinois/NCSA
Open Source License

Copyright (c) 2009-2015 by the contributors listed in CREDITS.TXT

All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

http://llvm.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal with the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.

* Neither the names of the LLVM Team, University of Illinois at Urbana-Champaign, nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE
SOFTWARE.

=====

Copyright (c) 2009-2014 by the contributors listed in CREDITS.TXT

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

===========================================================================

Mapbox GL uses portions of libcurl.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2015, Daniel Stenberg. <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright
notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER DEALINGS IN THE SOFTWARE.
Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

===========================================================================

Mapbox GL uses portions of libjpeg-turbo.

This software is based in part on the work of the Independent JPEG Group.

Copyright (C)2009-2015 D. R. Commander. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS", AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

TurboJPEG/LJT: this implements the TurboJPEG API using libjpeg or libjpeg-turbo

===========================================================================

Mapbox GL uses portions of libpng.

This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:
If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.

libpng versions 1.0.7, July 1, 2000, through 1.6.18, July 23, 2015, are Copyright (c) 2000-2002, 2004, 2006-2015 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

=================================

Mapbox GL uses portions of libuv.

libuv is part of the Node project: http://nodejs.org/
libuv may be distributed alone under Node's license:

=====

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

=====

This license applies to all parts of libuv that are not externally
maintained libraries.

The externally maintained libraries used by libuv are:

- tree.h (from FreeBSD), copyright Niels Provos. Two clause BSD license.

- inet_pton and inet_ntop implementations, contained in src/inet.c, are
copyright the Internet Systems Consortium, Inc., and licensed under the ISC
license.

- stdint-msvc2008.h (from msinttypes), copyright Alexander Chemeris. Three
clause BSD license.

- pthread-fixes.h, pthread-fixes.c, copyright Google Inc. and Sony Mobile
Communications AB. Three clause BSD license.

- android-ifaddrs.h, android-ifaddrs.c, copyright Berkeley Software Design
Inc, Kenneth MacKay and Emergya (Cloud4all, FP7/2007-2013, grant agreement
n 289016). Three clause BSD license.

===========================================================================

Mapbox GL uses portions of libzip.

Copyright (C) 1999-2014 Dieter Baron and Thomas Klausner

The authors can be contacted at <libzip@nih.at>
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===========================================================================

Mapbox GL uses portions of LOST.

Copyright (c) 2014 Mapzen

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

===========================================================================

Open Source Used In DNAC Platform Arica 6171
Mapbox GL uses portions of the Mapbox iOS SDK, which was derived from the Route-Me open source project, including the Alpstein fork of it.

The Route-Me license appears below.

Copyright (c) 2008-2013, Route-Me Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===========================================================================

Mapbox GL uses portions of nunicode.

Copyright (c) 2013 Aleksey Tulinov <aleksey.tulinov@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

===========================================================================

Mapbox GL uses portions of OkHTTP.

Copyright 2014 Square, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

===========================================================================

Mapbox GL uses portions of OpenSSL.

LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
All rights reserved.

This package is an SSL implementation written
by Eric Young (eay@cryptsoft.com).
The implementation was written so as to conform with Netscape SSL.

This library is free for commercial and non-commercial use as long as
The following conditions are aheared to. The following conditions
apply to all code found in this distribution, be it the RC4, RSA,
lhash, DES, etc., code; not just the SSL code. The SSL documentation
included with this distribution is covered by the same copyright terms
except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in
the code are not to be removed.
If this package is used in a product, Eric Young should be given attribution
as the author of the parts of the library used.
This can be in the form of a textual message at program startup or
in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must display the following acknowledgement:
   "This product includes cryptographic software written by
   Eric Young (eay@cryptsoft.com)"
   The word 'cryptographic' can be left out if the rouines from the library
   being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from
   the apps directory (application code) you must include an acknowledgement:
   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The licence and distribution terms for any publically available version or
derivative of this code cannot be changed. i.e. this code cannot simply be
copied and put under another distribution licence
[including the GNU Public Licence.]

===========================================================================

Mapbox GL uses portions of RapidJSON.

Tencent is pleased to support the open source community by making RapidJSON
available.

Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip. All rights
reserved.

If you have downloaded a copy of the RapidJSON binary from Tencent, please note
that the RapidJSON binary is licensed under the MIT License. If you have
downloaded a copy of the RapidJSON source code from Tencent, please note that
RapidJSON source code is licensed under the MIT License, except for the third-
party components listed below which are subject to different license terms.
Your integration of RapidJSON into your own projects may require compliance with
the MIT License, as well as the other licenses applicable to the third-party
components included within RapidJSON. To avoid the problematic JSON license in
your own projects, it's sufficient to exclude the bin/jsonchecker/ directory, as
it's the only code under the JSON license. A copy of the MIT License is included
in this file.

Other dependencies and licenses:

Open Source Software Licensed Under the BSD License:

The msinttypes r29
Copyright (c) 2006-2013 Alexander Chemeris
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
* Neither the name of copyright holder nor the names of its contributors may be
  used to endorse or promote products derived from this software without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Open Source Software Licensed Under the JSON License:

-----------------------------------------------

json.org
Copyright (c) 2002 JSON.org
All Rights Reserved.

JSON_checker
Copyright (c) 2002 JSON.org
All Rights Reserved.

Terms of the JSON License:
-----------------------------------------------

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Terms of the MIT License:
-----------------------------------------------

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

===========================================================================

Mapbox GL uses portions of Reachability.

Copyright (c) 2011, Tony Million.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===========================================================================

Mapbox GL uses portions of SQLite.
2001 September 15

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

    May you do good and not evil.
    May you find forgiveness for yourself and forgive others.
    May you share freely, never taking more than you give.

===========================================================================

Mapbox GL uses portions of SVPulsingAnnotationView.

Copyright (c) 2013, Sam Vermette <hello@samvermette.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

===========================================================================

Mapbox GL uses portions of zlib.

Acknowledgments:

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:

(C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

===========================================================================

Mapbox GL uses portions of Realm Objective-C.

Copyright 2015 Realm Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/****************************************************************************
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
**
** This file is part of the documentation of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:FDLS
** Commercial License Usage
** Licensees holding valid commercial Qt licenses may use this file in
** accordance with the commercial license agreement provided with the
** Software or, alternatively, in accordance with the terms contained in
** a written agreement between you and The Qt Company. For licensing terms
** and conditions see https://www.qt.io/terms-conditions. For further
** information use the contact form at https://www.qt.io/contact-us.
**
** GNU Free Documentation License Usage

Open Source Used In DNAC Platform Arica 6180
Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
ii. moral rights retained by the original author(s) and/or performer(s);

iii. publicity and privacy rights pertaining to a person’s image or likeness depicted in a Work;

iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;

v. rights protecting the extraction, dissemination, use and reuse of data in a Work;

vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and

vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.
a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

// Copyright 2015 The Chromium OS Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2014-2015 Calvin Metcalf and browserify-sign contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above
Copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License (MIT)

Copyright (c) Ben Drucker <bvdrucker@gmail.com> (bendrucker.me)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/****************************************************************************
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
**
** This file is part of the FOO module of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:LGPL3$
** Commercial License Usage
** Licensees holding valid commercial Qt licenses may use this file in
** accordance with the commercial license agreement provided with the
** Software or, alternatively, in accordance with the terms contained in
** a written agreement between you and The Qt Company. For licensing terms
** and conditions see http://www.qt.io/terms-conditions. For further
** information use the contact form at http://www.qt.io/contact-us.
**
** GNU Lesser General Public License Usage
** Alternatively, this file may be used under the terms of the GNU Lesser

---

Open Source Used In DNAC Platform Arica 6184
** General Public License version 3 as published by the Free Software
** Foundation and appearing in the file LICENSE.LGPLv3 included in the
** packaging of this file. Please review the following information to
** ensure the GNU Lesser General Public License version 3 requirements
** will be met: https://www.gnu.org/licenses/lgpl.html.
**
** GNU General Public License Usage
** Alternatively, this file may be used under the terms of the GNU
** General Public License version 2.0 or later as published by the Free
** Software Foundation and appearing in the file LICENSE.GPL included in
** the packaging of this file. Please review the following information to
** ensure the GNU General Public License version 2.0 requirements will be
** met: http://www.gnu.org/licenses/gpl-2.0.html.
**
** $QT_END_LICENSE$
**
****************************************************************************/

PLY (Python Lex-Yacc)                   Version 3.4

Copyright (C) 2001-2011,
David M. Beazley (Dabeaz LLC)
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
* Neither the name of the David Beazley or Dabeaz LLC may be used to
  endorse or promote products derived from this software without
  specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Jed Watson <jed.watson@me.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Mozilla Public License
Version 1.1

------------

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.
1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
   A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.
   B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1.
For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License. The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code. Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.
3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation
for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description
must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS,
WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING,
WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF
DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.
THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE
IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT,
YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER
OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF
ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken
8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.


11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County.
California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as “Multiple-Licensed”. “Multiple-Licensed” means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

```
```

The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is ______________________________________.
The Initial Developer of the Original Code is ________________________.
Portions created by ______________________ are Copyright (C) ______
______________________. All Rights Reserved.
Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[____] License"), in which case the provisions of [____] License are applicable instead of those
above. If you wish to allow use of your version of this file only under the terms of the [___] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License.”

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND OTHER CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR OTHER CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the author nor the names of other contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND OTHER CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR OTHER CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-------------------------------------------------------------------------------

NASM is now licensed under the 2-clause BSD license, also known as the simplified BSD license.

Copyright 1996-2009 the NASM Authors - All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2014-2018 Omar Cornut

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

* May you do good and not evil.
* May you find forgiveness for yourself and forgive others.
* May you share freely, never taking more than you give.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you

Open Source Used In DNAC Platform Arica 6198
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULDN'T THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) <year> <name of author>  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

The MIT License

Copyright (c) 2013 Max Ogden

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2016 Guillermo Rauch (@rauchg)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2013-2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2017 Garen J. Torikian

MIT License
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011-2014 TJ Holowaychuk
Copyright (c) 2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# Names should be added to this file as:
#     Name <email address>

David Beaumont <dbeaumont@google.com>
David Yonge-Mallo <davinci@google.com>
Dong Zhou <dongzhou@google.com>
Fredrik Roubert <roubert@google.com>
Jeanine Lilleng <jeanine@google.com>
Keghani Kouzoujian <keghani@google.com>
Lara Scheidegger <lararennie@google.com>
Rouslan Solomakhin <rouslan@chromium.org>
Shaopeng Jia <shaopengjia@google.com>

(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License (MIT))

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 1987, 1988 X Consortium

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

Copyright 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts.

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Digital not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

DIGITAL DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Copyright (c) 2001 Robert Penner
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
* Neither the name of the author nor the names of contributors may be used
  to endorse or promote products derived from this software without specific
  prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

node-http-proxy

Copyright (c) 2010-2016 Charlie Robbins, Jarrett Cruger & the Contributors.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.


Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a ”NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to
the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright (C) 2012 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY APPLE, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (c) spdx-satisfies.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar license but with different Copyright notices) all the files are:

Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2018 Toru Nagashima

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2015 Gareth Jones (with contributions from many other people)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Condition of use and distribution are the same as zlib:

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgement in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Copyright (c) 2013-2014 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation’s software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

// Copyright (c) 2012 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010 passive.ly LLC

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
Open Source Used In DNAC Platform Arica  6225

included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
The libarchive distribution as a whole is Copyright by Tim Kientzle
and is subject to the copyright notice reproduced at the bottom of
this file.

Each individual file in this distribution should have a clear
copyright/licensing statement at the beginning of the file. If any do
not, please let me know and I will rectify it. The following is
intended to summarize the copyright status of the individual files;
the actual statements in the files are controlling.

* Except as listed below, all C sources (including .c and .h files)
and documentation files are subject to the copyright notice reproduced
at the bottom of this file.

* The following source files are also subject in whole or in part to
a 3-clause UC Regents copyright; please read the individual source
files for details:
  libarchive/archive_entry.c
  libarchive/archive_read_support_compression_compress.c
  libarchive/archive_write_set_compression_compress.c
  libarchive/mtree.5
  tar/matching.c

* The following source files are in the public domain:
  tar/getdate.c

* The build files---including Makefiles, configure scripts,
and auxiliary scripts used as part of the compile process---have
widely varying licensing terms. Please check individual files before
distributing them to see if those restrictions apply to you.

I intend for all new source code to use the license below and hope over
time to replace code with other licenses with new implementations that
do use the license below. The varying licensing of the build scripts
seems to be an unavoidable mess.

Copyright (c) 2003-2009 <author(s)>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer in this position and unchanged.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR(S) "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2012 Colingo.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:
Sanjay Ghemawat <sanjay@google.com>
Jeff Dean <jeff@google.com>
Daniel Dulitz <daniel@google.com>
Craig Silverstein
Paul Haahr <haahr@google.com>
Corey Anderson <corin@google.com>
(and many others)

Proto2 C++ and Java primary author:
Kenton Varda <kenton@google.com>

Proto2 Python primary authors:
Will Robinson <robinson@google.com>
Petar Petrov <petar@google.com>

Java Nano primary authors:
Brian Duff <bduff@google.com>
Tom Chao <chaot@google.com>
Max Cai <maxtroy@google.com>
Ulas Kirazci <ulas@google.com>

Large code contributions:
Jason Hsueh <jasonh@google.com>
Joseph Schorr <jschorr@google.com>
Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:
Scott Bruce <sbruce@google.com>
Frank Yellin
Neal Norwitz <nnorwitz@google.com>
Jeffrey Yasskin <jyasskin@google.com>
Ambrose Feinstein <ambrose@google.com>

Documentation:
Lisa Carey <lcarey@google.com>

Maven packaging:
Gregory Kick <gak@google.com>

Patch contributors:
Kevin Ko <kevin.s.ko@gmail.com>
  * Small patch to handle trailing slashes in --proto_path flag.
Johan Euphrosine <proppy@aminche.com>
  * Small patch to fix Python CallMethod().
Ulrich Kunitz <kune@deine-taler.de>
  * Small optimizations to Python serialization.
Leandro Lucarella <llucax@gmail.com>
  * VI syntax highlighting tweaks.
  * Fix compiler to not make output executable.
Dilip Joseph <dilip.antony.joseph@gmail.com>
  * Heuristic detection of sub-messages when printing unknown fields in
Open Source Used In DNAC Platform Arica

text format.

Brian Atkinson <bnaibr774@gmail.com>
* Added @Override annotation to generated Java code where appropriate.

Vincent Choinire <Choinire.Vincent@hydro.qc.ca>
* Tru64 support.

Monty Taylor <monty.taylor@gmail.com>
* Solaris 10 + Sun Studio fixes.

Alek Storm <alek.storm@gmail.com>
* Slicing support for repeated scalar fields for the Python API.

Oleg Smolsky <oleg.smolsky@gmail.com>
* MS Visual Studio error format option.
  * Detect unordered_map in stl_hash.m4.

Brian Olson <brianolson@google.com>
* gzip/zlib I/O support.

Michael Poole <mdpoole@troilus.org>
* Fixed warnings about generated constructors not explicitly initializing all fields (only present with certain compiler settings).
* Added generation of field number constants.

Wink Saville <wink@google.com>
* Fixed initialization ordering problem in logging code.

Will Pierce <willp@nuclei.com>
* Small patch improving performance of in Python serialization.

Alexandre Vassalotti <alexandre@peadrop.com>
* Emacs mode for Protocol Buffers (editors/protobuf-mode.el).

Scott Stafford <scott.stafford@gmail.com>
* Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.

Alexander Melnikov <alm@sibmail.ru>
* HPUX support.

Oliver Jowett <oliver.jowett@gmail.com>
* Detect whether zlib is new enough in configure script.
  * Fixes for Solaris 10 32/64-bit confusion.

Evan Jones <evanj@mit.edu>
* Optimize Java serialization code when writing a small message to a stream.
  * Optimize Java serialization of strings so that UTF-8 encoding happens only once per string per serialization call.
  * Clean up some Java warnings.
  * Fix bug with permanent callbacks that delete themselves when run.

Michael Kucharski <m.kucharski@gmail.com>
* Added CodedInputStream.getTotalBytesRead().

Kacper Kowalik <xarthisius.kk@gmail.com>
* Fixed m4/acx_pthread.m4 problem for some Linux distributions.

William Orr <will@worrbase.com>
* Fixed detection of sched_yield on Solaris.
  * Added atomicops for Solaris

Andrew Paprocki <andrew@ishiboo.com>
* Fixed minor IBM xC compiler build issues
  * Added atomicops for AIX (POWER)
The MIT License

Copyright (C) 2011-2019 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (c) 2010 Timothy Farrell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2017, The CRC32C Authors.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.

* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2017 Samuel Reed <samuel.trace.reed@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Licensed under the Academic Free License version 2.0 (below)

Or under the following terms:

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

--------------------------------------------------------------------------------

Academic Free License v. 2.0

--------------------------------------------------------------------------------

This Academic Free License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following notice immediately following the copyright notice for the Original Work:

Licensed under the Academic Free License version 2.0
1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:

a) to reproduce the Original Work in copies;
b) to prepare derivative works ("Derivative Works") based upon the Original Work;
c) to distribute copies of the Original Work and Derivative Works to the public;
d) to perform the Original Work publicly; and
e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license, under patent claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.
4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to license.

5) This section intentionally omitted.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.
9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries, and by international treaty. Therefore, by exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, for patent infringement (i) against Licensor with respect to a patent applicable to software or (ii) against any entity with respect to a patent applicable to the Original Work (but excluding combinations of the Original Work with other software or hardware).

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. 101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys’ fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.

13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You” includes any entity that controls, is controlled by, or is
under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.

This license is Copyright (C) 2003 Lawrence E. Rosen. All rights reserved. Permission is hereby granted to copy and distribute this license without modification. This license may not be modified without the express written permission of its copyright owner.

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium: https://cla.developers.google.com/clas

Files in third_party/ have their own licenses, as described therein. The MIT license, for third_party/fiat, which, unlike other third_party directories, is compiled into non-test libraries, is included below.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

The following are Google-internal bug numbers where explicit permission from some authors is recorded for use of their work. (This is purely for our own record keeping.)

27287199
27287880
27287883

OpenSSL License

-------------

/* ====================================================================
 * Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.
 * *
 * Redistribution and use in source and binary forms, with or without *
 * modification, are permitted provided that the following conditions *
 * are met:
 * *
 */
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
   
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
   
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).
/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *    "This product includes cryptographic software written by
 *     Eric Young (eay@cryptsoft.com)"
 *    The word 'cryptographic' can be left out if the rouines from the library
 *    being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 *    the apps directory (application code) you must include an acknowledgement:
 *    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
The code in third_party/fiat carries the MIT license:

Copyright (c) 2015-2016 the fiat-crypto authors (see https://github.com/mit-plv/fiat-crypto/blob/master/AUTHORS).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Licenses for support code
--------------------------

Parts of the TLS test suite are under the Go license. This code is not included
in BoringSSL (i.e. libcrypto and libssl) when compiled, however, so
distributing code linked against BoringSSL does not trigger this license:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BoringSSL uses the Chromium test infrastructure to run a continuous build,
trybots etc. The scripts which manage this, and the script for generating build
metadata, are under the Chromium license. Distributing code linked against
BoringSSL does not trigger this license.

Copyright 2015 The Chromium Authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
# This is the official list of people who can contribute
# (and typically have contributed) code to the RE2 repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# The submission process automatically checks to make sure
# that people submitting code are listed in this file (by email address).
#
# Names should be added to this file only after verifying that
# the individual or the individual's organization has agreed to
# the appropriate Contributor License Agreement, found here:
#
#   http://code.google.com/legal/individual-cla-v1.0.html
#
# The agreement for individuals can be filled out on the web.
#
# When adding J Random Contributor's name to this file,
# either J's name or J's organization's name should be
# added to the AUTHORS file, depending on whether the
# individual or corporate CLA was used.
#
# Names should be added to this file like so:
#   Name <email address>
#
# Please keep the list sorted.

Dominic Battr <battre@chromium.org>
Doug Kwan <doukgwan@google.com>
Dmitriy Vyukov <dvyukov@google.com>
John Millkin <jmillkin@gmail.com>
Mike Nazarewicz <mpn@google.com>
Nico Weber <thakis@chromium.org>
Pawel Hajdan <phajdan.jr@gmail.com>
Rob Pike <r@google.com>
Russ Cox <rsc@swtch.com>
Sanjay Ghemawat <sanjay@google.com>
Stefano Rivera <stefano.rivera@gmail.com>
Srinivasan Venkatachary <vsri@google.com>
Viatcheslav Ostapenko <sl.ostapenko@samsung.com>

MIT License
Copyright (c) 2012-2014 Chris Pettitt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2017 Matt Lewis

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2006, 2008 Edward Rosten
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the University of Cambridge nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/
* Copyright(c)1995,97 Mark Olesen <olesen@me.QueensU.CA>
* Queen's Univ at Kingston (Canada)
*
* Permission to use, copy, modify, and distribute this software for
* any purpose without fee is hereby granted, provided that this
* entire notice is included in all copies of any software which is
* or includes a copy or modification of this software and in all
* copies of the supporting documentation for such software.
*
* THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR
* IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR QUEEN'S
* UNIVERSITY AT KINGSTON MAKES ANY REPRESENTATION OR WARRANTY OF ANY
* KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS
* FITNESS FOR ANY PARTICULAR PURPOSE.
*
* All of which is to say that you can do what you like with this
* source code provided you don't try to sell it as your own and you
* include an unaltered copy of this message (including the
* copyright).
*
* It is also implicitly understood that bug fixes and improvements
* should make their way back to the general Internet community so
* that everyone benefits.
*/
Notice that the following BSD-style license applies to this one file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

This file is part of Valgrind, a dynamic binary instrumentation framework.

Copyright (C) 2000-2007 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR `AS IS` AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2012 Yahoo Inc.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# This is the official list of people who can contribute
# (and who have contributed) code to the ANGLE project
# repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#

TransGaming Inc.
Nicolas Capens
Daniel Koch
Geoff Lang
Andrew Lewycky
Jamie Madill
Gavriel State
Shannon Woods

Google Inc.
Brent Austin
Michael Bai
John Bauman
Peter Beverloo
Steve Block
Rachel Blum
Eric Boren
Henry Bridge
Nat Duca
Courtney Goeltzenleuchter
Peter Kasting
Vangelis Kokkevis
Zhenyao Mo
Daniel Nicoara
Alastair Patrick
Alok Priyadarshi
Kenneth Russell
Brian Salomon
Gregg Tavares
Jeff Timanus
Ben Vanik
Adrienne Walker
thestig@chromium.org
Justin Schuh
Scott Graham
Corentin Wallez
Kai Ninomiya
Victor Costan
Shahbaz Youssefi

Adobe Systems Inc.
Alexandru Chiculita
Steve Minns
Max Vujovic

Autodesk, Inc.
Ranger Harke

Cloud Party, Inc.
Conor Dickinson

The Qt Company Ltd.
Andrew Knight

Imagination Technologies Ltd.
Gregoire Payen de La Garanderie

Intel Corporation
Jin Yang
Andy Chen
Josh Triplett
Sudarsana Nagineni
Jiajia Qin
Jiawei Shao
Jie Chen
Qiankun Miao
Bryan Bernhart
Yunchao He
Xinghua Cao
Brandon Jones
Enrico Galli

Klarlvdalens Datakonsult AB
Milian Wolff

Mozilla Corp.
Dzmitry Malyshau
Ehsan Akhgari
Edwin Flores
Jeff Gilbert
Mike Hommey
Benoit Jacob
Makoto Kato
Vladimir Vukicevic
Josh Matthews

Turbulenz
Michael Braithwaite

Ulrik Persson (ddefrosst)
Mark Banner (standard8mbp)
David Kilzer
Jacek Caban
Tibor den Ouden
Rgis Fnon
Sebastian Bergstein
James Ross-Gowan
Andrei Volykhin
Jrme Duval
Till Rathmann
Nick Shaforostov

Microsoft Corporation
Cooper Partin
Austin Kinross
Mingyu Hu
Minmin Gong
Shawn Hargreaves
Rafael Cintron
James Clarke
Nathan Zabriskie

Microsoft Open Technologies, Inc.
Cooper Partin
Austin Kinross
NVIDIA Corporation
Olli Etuaho
Arun Patole
Qingqing Deng
Kimmo Kinnunen
Sami Viisnen
Martin Radev
Joonatan Saarhelo
Markus Tavenrath
Anders Leino

Opera Software ASA
Daniel Bratell
Tomasz Moniuszko
David Landell

Advanced Micro Devices, Inc.
Russ Lind

LG Electronics, Inc.
Jani Hautakangas

IBM Inc.
Junliang Yan

AdaptVis GmbH
Sascha Kolodzey

Samsung Electronics, Inc.
Brandon Schade
Colin Peter
Hyunchang Kim
Hyunseok Ko
Jaedon Lee
Jeff Vigil
Kevin Jung
Kyeongmin Kim
Minkyu Jeong
Mohan Maiya
Sangwon Park

Arm Ltd.
Fei Yang
Xinyi He
Sunny Sun

GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below. This license makes reference to the version 3 of the GNU General Public License, which you can find in the LICENSE.GPLv3 file.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this licencedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, this License refers to version 3 of the GNU Lesser General Public License, and the GNU GPL refers to version 3 of the GNU General Public License.

The Library refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An Application is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A Combined Work is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the Linked Version.

The Minimal Corresponding Source for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The Corresponding Application Code for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.
1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

   a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

   b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

   a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

   b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

   a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library
facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License or any later version applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

/*
 * Written by Wilco Dijkstra, 1996. The following email exchange establishes the license.
 *
 * From: Wilco Dijkstra <Wilco.Dijkstra@ntlworld.com>
 * Date: Fri, Jun 24, 2011 at 3:20 AM
 * Subject: Re: sqrt routine
 * To: Kevin Ma <kma@google.com>
 * Hi Kevin,
 * Thanks for asking. Those routines are public domain (originally posted to
 * comp.sys.arm a long time ago), so you can use them freely for any purpose.
 * Cheers,
 * Wilco
 *
 * ----- Original Message ----- 
 * From: "Kevin Ma" <kma@google.com>
 * To: <Wilco.Dijkstra@ntlworld.com>
 * Sent: Thursday, June 23, 2011 11:44 PM
 * Subject: Fwd: sqrt routine
 * Hi Wilco,
 * I saw your sqrt routine from several web sites, including
Mozilla Public License, version 2.0

1. Definitions

1.1. "Contributor"

  means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

  means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

  means Covered Software of a particular Contributor.

1.4. "Covered Software"

  means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

  means

  a. that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

  b. that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

  means any form of the work other than Source Code Form.

1.7. "Larger Work"

  means a work that combines Covered Software with other material, in a
separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

a. any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

b. any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.
2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

a. under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

b. under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

a. for any code that a Contributor has removed from Covered Software; or

b. for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

c. under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses
No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

a. such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

b. You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the
license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.
5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any
character including, without limitation, damages for lost profits, loss of
 goodwill, work stoppage, computer failure or malfunction, or any and all
 other commercial damages or losses, even if such party shall have been
 informed of the possibility of such damages. This limitation of liability
 shall not apply to liability for death or personal injury resulting from
 such party’s negligence to the extent applicable law prohibits such
 limitation. Some jurisdictions do not allow the exclusion or limitation of
 incidental or consequential damages, so this exclusion and limitation may
 not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts
 of a jurisdiction where the defendant maintains its principal place of
 business and such litigation shall be governed by laws of that
 jurisdiction, without reference to its conflict-of-law provisions. Nothing
 in this Section shall prevent a party’s ability to bring cross-claims or
 counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject
 matter hereof. If any provision of this License is held to be
 unenforceable, such provision shall be reformed only to the extent
 necessary to make it enforceable. Any law or regulation which provides that
 the language of a contract shall be construed against the drafter shall not
 be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
 10.3, no one other than the license steward has the right to modify or
 publish new versions of this License. Each version will be given a
 distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
 of the License under which You originally received the Covered Software,
 or under the terms of any subsequent version published by the license
 steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/. If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0. The code in lib/flat_buffers.dart is based on code that was released under the following license:

Copyright 2012, the Dart project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

To the extent permissible, the changes to that code and the other assets in this package are licensed under the Apache2 license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2014 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright 2013 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (C) 2013 Vojta Jna.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (C) 2011-2015 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (C) 2014 Wei Fanzhe
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright 2019 The Fuchsia Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:

// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 1999 Serika Kurusugawa. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
Copyright (c) 2017, crypto-browserify contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

W3C SOFTWARE NOTICE AND LICENSE

This license came from: https://www.w3.org/Consortium/Legal/2015/copyright-software-and-document

This work is being provided by the copyright holders under the following license.

License
By obtaining and/or copying this work, you (the licensee) agree that you have read, understood, and will comply
with the following terms and conditions.

Permission to copy, modify, and distribute this work, with or without modification, for any purpose and without fee
or royalty is hereby granted, provided that you include the following on ALL copies of the work or portions thereof,
including modifications:

- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C
  Software and Document Short Notice should be included.
- Notice of any changes or modifications, through a copyright statement on the new code or document such as "This
  software or document includes material copied from or derived from [title and URI of the W3C document]."
Copyright [YEAR] W3C (MIT, ERCIM, Keio, Beihang).

Disclaimers

THIS WORK IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENT WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENT.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the work without specific, written prior permission. Title to copyright in this work will at all times remain with copyright holders.

NOTE! The LGPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the kde libraries) are copyrighted by the authors who actually wrote it.

---------------------------------------------------------------------------

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1301, USA.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in
a textual and legal sense, the linked executable is a combined work, a
derivative of the original library, and the ordinary General Public License
treats it as such.

Because of this blurred distinction, using the ordinary General
Public License for libraries did not effectively promote software
sharing, because most developers did not use the libraries. We
concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the
users of those programs of all benefit from the free status of the
libraries themselves. This Library General Public License is intended to
permit developers of non-free programs to use free libraries, while
preserving your freedom as a user of such programs to change the free
libraries that are incorporated in them. (We have not seen how to achieve
this as regards changes in header files, but we have achieved it as regards
changes in the actual functions of the Library.) The hope is that this
will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, while the latter only
works together with the library.

Note that it is possible for a library to be covered by the ordinary
General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which
contains a notice placed by the copyright holder or other authorized
party saying it may be distributed under the terms of this Library
General Public License (also called "this License"). Each licensee is
addressed as "you".

A "library" means a collection of software functions and/or data
prepared so as to be conveniently linked with application programs
(which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the
Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a
portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is
included without limitation in the term "modification".)
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.
(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

   However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

   When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

   If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

   Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

   You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Library General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, 
but WITHOUT ANY WARRANTY; without even the implied warranty of 
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU 
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public 
License along with this library; if not, write to the Free Software 
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your 
school, if any, to sign a "copyright disclaimer" for the library, if 
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the 
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<stdio of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
MIT License

Copyright (c) 2016 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all 
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE 
SOFTWARE.

****************************************************************************
**
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
This file is part of the FOO module of the Qt Toolkit.

GNU Lesser General Public License Usage

This file may be used under the terms of the GNU Lesser
General Public License version 2.1 as published by the Free Software
Foundation and appearing in the file LICENSE.LGPL included in the
packaging of this file. Please review the following information to
ensure the GNU Lesser General Public License version 2.1 requirements
will be met: http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html.

If you have questions regarding the use of this file, please contact
us via http://www.qt.io/contact-us/.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and/or associated documentation files (the
"Materials"), to deal in the Materials without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Materials, and to
permit persons to whom the Materials are furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Materials.

MODIFICATIONS TO THIS FILE MAY MEAN IT NO LONGER ACCURATELY REFLECTS
KHRONOS STANDARDS. THE UNMODIFIED, NORMATIVE VERSIONS OF KHRONOS
SPECIFICATIONS AND HEADER INFORMATION ARE LOCATED AT
https://www.khronos.org/registry/

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
MIT License

Copyright (c) 2019 Sindresorhus <sindresorhus@gmail.com> (https://sindresorhus.com), Paul Miller
(https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2011 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
    distribution, then any Derivative Works that You distribute must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file distributed
    as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2014, Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

amdefine is released under two licenses: new BSD, and MIT. You may pick the license that best suits your development needs. The text of both licenses are provided below.

The "New" BSD License:

Copyright (c) 2011-2016, The Dojo Foundation
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License
----------

Copyright (c) 2011-2016, The Dojo Foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2012-2018 Luke Plaster <notatestuser@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
// Copyright 2018 The Chromium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
Copyright (c) 2007-2010 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Materials, and to 
permit persons to whom the Materials are furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be included 
in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. 
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY 
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, 
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE 
MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

SGI FREE SOFTWARE LICENSE B (Version 2.0, Sept. 18, 2008)

Copyright (C) 1992 Silicon Graphics, Inc. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of 
this software and associated documentation files (the "Software"), to deal in 
the Software without restriction, including without limitation the rights to 
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies 
of the Software, and to permit persons to whom the Software is furnished to do 
so, subject to the following conditions:

The above copyright notice including the dates of first publication and either 
this permission notice or a reference to http://oss.sgi.com/projects/FreeB/ 
shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL SILICON 
GRAPHICS, INC. BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN 
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Silicon Graphics, Inc. shall 
not be used in advertising or otherwise to promote the sale, use or other 
dealings in this Software without prior written authorization from Silicon 
Graphics, Inc. 
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person 
obtaining a copy of this software and associated documentation 
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, 
copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the 
Software is furnished to do so, subject to the following 
conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES 
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT 
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, 
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING 
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR 
OTHER DEALINGS IN THE SOFTWARE.
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any 
purpose with or without fee is hereby granted, provided that the above 
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES 
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF 
MERCHANTABILITY AND FITNESS. IN NO EVENT WILL THE AUTHOR BE LIABLE FOR 
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES 
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN 
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR 
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## Glob Logo

Glob's logo created by Tanya Brassie <http://tanyabrassie.com/>, licensed 
under a Creative Commons Attribution-ShareAlike 4.0 International License 
https://creativecommons.org/licenses/by-sa/4.0/
'zlib' general purpose compression library 
version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied 
warranty. In no event will the authors be held liable for any damages 
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu
Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
(The MIT License)

Copyright (c) 2014-2015 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

# The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Clipper code library, the "Software" (that includes Delphi, C++ & C# source code, accompanying samples and documentation), has been released under the following license, terms and conditions:

Boost Software License - Version 1.0 - August 17th, 2003
http://www.boost.org/LICENSE_1_0.txt

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2016 Kirill Fomichev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (C) 2000-2017 Julian Seward. All rights reserved.
Copyright (C) 2003-2017 Josef Weidendorfer. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must
   not claim that you wrote the original software. If you use this
   software in a product, an acknowledgment in the product
   documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2007-2008, Apple, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Apple, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The ISC License

Copyright (c) 2015, 2019 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2005, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ESLint
Copyright jQuery Foundation and other contributors, https://jquery.org/

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Redistribution and use of this software in source and binary forms,
with or without modification, are permitted provided that the following
conditions are met:

* Redistributions of source code must retain the above
  copyright notice, this list of conditions and the
  following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the
  following disclaimer in the documentation and/or other
  materials provided with the distribution.

* Neither the name of the assimp team, nor the names of its
  contributors may be used to endorse or promote products
  derived from this software without specific prior
  written permission of the assimp team.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

If you want to write an option parser, and have it be good, there are
two ways to do it. The Right Way, and the Wrong Way.
The Wrong Way is to sit down and write an option parser. We've all done that.

The Right Way is to write some complex configurable program with so many options that you hit the limit of your frustration just trying to manage them all, and defer it with duct-tape solutions until you see exactly to the core of the problem, and finally snap and write an awesome option parser.

If you want to write an option parser, don't write an option parser. Write a package manager, or a source control system, or a service restarter, or an operating system. You probably won't end up with a good one of those, but if you don't give up, and you are relentless and diligent enough in your procrastination, you may just end up with a very nice option parser.

## USAGE

```javascript
// my-program.js
var nopt = require("nopt")
, Stream = require("stream").Stream
, path = require("path")
, knownOpts = { "foo" : [String, null]
, "bar" : [Stream, Number]
, "baz" : path
, "bloo" : [ "big", "medium", "small" ]
, "flag" : Boolean
, "pick" : Boolean
, "many1" : [String, Array]
, "many2" : [path, Array]
}
, shortHands = { "foofoo" : ["--foo", "Mr. Foo"]
, "b7" : ["--bar", "7"]
, "m" : ["--bloo", "medium"]
, "p" : ["--pick"]
, "f" : ["--flag"]
}

// everything is optional.
// knownOpts and shorthands default to {}
// arg list defaults to process.argv
// slice defaults to 2
, parsed = nopt(knownOpts, shortHands, process.argv, 2)
console.log(parsed)
```

This would give you support for any of the following:
```console
$ node my-program.js --foo "blerp" --no-flag
{ "foo" : "blerp", "flag" : false }

$ node my-program.js ---bar 7 --foo "Mr. Hand" --flag
{ bar: 7, foo: "Mr. Hand", flag: true }

$ node my-program.js --foo "blerp" -f ----p
{ foo: "blerp", flag: true, pick: true }

$ node my-program.js -fp --foofoo
{ foo: "Mr. Foo", flag: true, pick: true }

$ node my-program.js --foofoo --fp # -- stops the flag parsing.
{ foo: "Mr. Foo", argv: [ remain: ["-fp"] ] }

$ node my-program.js --blatzk -fp # unknown opts are ok.
{ blatzk: true, flag: true, pick: true }

$ node my-program.js --blatzk=1000 -fp # but you need to use = if they have a value
{ blatzk: 1000, flag: true, pick: true }

$ node my-program.js --no-blatzk -fp # unless they start with "no-"
{ blatzk: false, flag: true, pick: true }

$ node my-program.js --baz b/a/z # known paths are resolved.
{ baz: "/Users/isaacs/b/a/z" }

# if Array is one of the types, then it can take many
# values, and will always be an array. The other types provided
# specify what types are allowed in the list.

$ node my-program.js --many1 5 --many1 null --many1 foo
{ many1: ["5", "null", "foo"] }

$ node my-program.js --many2 foo --many2 bar
{ many2: ["/path/to/foo", "path/to/bar"] }

```

Read the tests at the bottom of `lib/nopt.js` for more examples of what this puppy can do.

## Types

The following types are supported, and defined on `nopt.typeDefs`

* String: A normal string. No parsing is done.
* path: A file system path. Gets resolved against cwd if not absolute.
* url: A url. If it doesn't parse, it isn't accepted.
* Number: Must be numeric.
* Date: Must parse as a date. If it does, and `Date` is one of the options, then it will return a Date object, not a string.
* Boolean: Must be either `true` or `false`. If an option is a boolean, then it does not need a value, and its presence will imply `true` as the value. To negate boolean flags, do `--no-whatever` or `--whatever false`.
* NaN: Means that the option is strictly not allowed. Any value will fail.
* Stream: An object matching the "Stream" class in node. Valuable for use when validating programmatically. (npm uses this to let you supply any WriteStream on the `outfd` and `logfd` config options.)
* Array: If `Array` is specified as one of the types, then the value will be parsed as a list of options. This means that multiple values can be specified, and that the value will always be an array.

If a type is an array of values not on this list, then those are considered valid values. For instance, in the example above, the `--bloo` option can only be one of "big", "medium", or "small", and any other value will be rejected.

When parsing unknown fields, "true", "false", and "null" will be interpreted as their JavaScript equivalents.

You can also mix types and values, or multiple types, in a list. For instance `{ blah: [Number, null] }` would allow a value to be set to either a Number or null. When types are ordered, this implies a preference, and the first type that can be used to properly interpret the value will be used.

To define a new type, add it to `nopt.typeDefs`. Each item in that hash is an object with a `type` member and a `validate` method. The `type` member is an object that matches what goes in the type list. The `validate` method is a function that gets called with `validate(data, key, val)`. Validate methods should assign `data[key]` to the valid value of `val` if it can be handled properly, or return boolean `false` if it cannot.

You can also call `nopt.clean(data, types, typeDefs)` to clean up a config object and remove its invalid properties.

## Error Handling

By default, nopt outputs a warning to standard error when invalid values for known options are found. You can change this behavior by assigning a method to `nopt.invalidHandler`. This method will be called with the offending `nopt.invalidHandler(key, val, types)`.
If no `nopt.invalidHandler` is assigned, then it will `console.error` its whining. If it is assigned to boolean ‘false’ then the warning is suppressed.

## Abbreviations

Yes, they are supported. If you define options like this:

```javascript
{ "foolhardyelephants" : Boolean , "pileofmonkeys" : Boolean }
```

Then this will work:

```bash
node program.js --foolhar --pil
node program.js --no-f --pileofmon
# etc.
```

## Shorthands

Shorthands are a hash of shorter option names to a snippet of args that they expand to.

If multiple one-character shorthands are all combined, and the combination does not unambiguously match any other option or shorthand, then they will be broken up into their constituent parts. For example:

```json
{ "s" : ["--loglevel", "silent"] , "g" : "--global" , "f" : "--force" , "p" : "--parseable" , "l" : "--long" }
```

```bash
npm ls -sgflp
# just like doing this:
npm ls --loglevel silent --global --force --long --parseable
```

## The Rest of the args
The config object returned by nopt is given a special member called `argv`, which is an object with the following fields:

* `remain`: The remaining args after all the parsing has occurred.
* `original`: The args as they originally appeared.
* `cooked`: The args after flags and shorthands are expanded.

## Slicing

Node programs are called with more or less the exact argv as it appears in C land, after the v8 and node-specific options have been plucked off. As such, `argv[0]` is always `node` and `argv[1]` is always the JavaScript program being run.

That's usually not very useful to you. So they're sliced off by default. If you want them, then you can pass in `0` as the last argument, or any other number that you'd like to slice off the start of the list.

MIT License

Copyright (c) 2016, 2018 Linus Unnebck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2012 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the

Open Source Used In DNAC Platform Arica 6306
Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

libjpeg-turbo Licenses

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

- The IJG (Independent JPEG Group) License, which is listed in
  [README.ijg](README.ijg)

This license applies to the libjpeg API library and associated programs
(any code inherited from libjpeg, and any modifications to that code.)
- The Modified (3-clause) BSD License, which is listed below

This license covers the TurboJPEG API library and associated programs, as well as the build system.

- The [zlib License](https://opensource.org/licenses/Zlib)

This license is a subset of the other two, and it covers the libjpeg-turbo SIMD extensions.

**Complying with the libjpeg-turbo Licenses**

This section provides a roll-up of the libjpeg-turbo licensing terms, to the best of our understanding.

1. If you are distributing a modified version of the libjpeg-turbo source, then:

   1. You cannot alter or remove any existing copyright or license notices from the source.

      **Origin**
      - Clause 1 of the IJG License
      - Clause 1 of the Modified BSD License
      - Clauses 1 and 3 of the zlib License

   2. You must add your own copyright notice to the header of each source file you modified, so others can tell that you modified that file (if there is not an existing copyright header in that file, then you can simply add a notice stating that you modified the file.)

      **Origin**
      - Clause 1 of the IJG License
      - Clause 2 of the zlib License

   3. You must include the IJG README file, and you must not alter any of the copyright or license text in that file.

      **Origin**
      - Clause 1 of the IJG License

2. If you are distributing only libjpeg-turbo binaries without the source, or if you are distributing an application that statically links with libjpeg-turbo, then:
1. Your product documentation must include a message stating:

   This software is based in part on the work of the Independent JPEG
   Group.

   **Origin**
   - Clause 2 of the IJG license

2. If your binary distribution includes or uses the TurboJPEG API, then
   your product documentation must include the text of the Modified BSD
   License (see below.)

   **Origin**
   - Clause 2 of the Modified BSD License

3. You cannot use the name of the IJG or The libjpeg-turbo Project or the
   contributors thereof in advertising, publicity, etc.

   **Origin**
   - IJG License
   - Clause 3 of the Modified BSD License

4. The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be
   free of defects, nor do we accept any liability for undesirable
   consequences resulting from your use of the software.

   **Origin**
   - IJG License
   - Modified BSD License
   - zlib License

The Modified (3-clause) BSD License
====================================

Copyright (C)2009-2019 D. R. Commander. All Rights Reserved.
Copyright (C)2015 Viktor Szathmry. All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its
  contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS", AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Why Three Licenses?
===================
The zlib License could have been used instead of the Modified (3-clause) BSD License, and since the IJG License effectively subsumes the distribution conditions of the zlib License, this would have effectively placed libjpeg-turbo binary distributions under the IJG License. However, the IJG License specifically refers to the Independent JPEG Group and does not extend attribution and endorsement protections to other entities. Thus, it was desirable to choose a license that granted us the same protections for new code that were granted to the IJG for code derived from their software.

// Copyright 2018 The ANGLE Project Authors.
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
//
// Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
//
// Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following
// disclaimer in the documentation and/or other materials provided
// with the distribution.
//
// Neither the name of TransGaming Inc., Google Inc., 3DLabs Inc.
// Ltd., nor the names of their contributors may be used to endorse
// or promote products derived from this software without specific
// prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT

The MIT License (MIT)
Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The MIT License (MIT)
Copyright (c) 2018 Made With MOXY Lda <hello@moxy.studio>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 hemanth

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2011-2018 JS Foundation and contributors, https://js.foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013 Alex Seville

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/* Copyright (c) 2008-2009, Google Inc.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are
 * met:
 * *
 * * Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * * Neither the name of Google Inc. nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission. */
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
* OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
* ---
* Author: Kostya Serebryany
*/

Copyright 2011 Kevin Burke unless otherwise noted.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Some content is copyrighted by Twitter, Inc., and also released under an
Apache License; these sections are noted in the source.
# This is the official list of people who can contribute
# (and who have contributed) code to the ANGLE project
# repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#

TransGaming Inc.
Nicolas Capens
Daniel Koch
Geoff Lang
Andrew Lewycky
Jamie Madill
Gavriel State
Shannon Woods
Google Inc.
Brent Austin
Michael Bai
John Bauman
Peter Beverloo
Steve Block
Rachel Blum
Eric Boren
Henry Bridge
Nat Duca
Peter Kasting
Vangelis Kokkevis
Zhenyao Mo
Daniel Nicoara
Alastair Patrick
Alok Priyadarshi
Kenneth Russell
Brian Salomon
Gregg Tavares
Jeff Timanus
Ben Vanik
Adrienne Walker
thestig@chromium.org
Justin Schuh
Scott Graham
Corentin Wallez
Kai Ninomiya

Adobe Systems Inc.
Alexandru Chiculita
Steve Minns
Max Vujovic

Autodesk, Inc.
Ranger Harke

Cloud Party, Inc.
Conor Dickinson

The Qt Company Ltd.
Andrew Knight

Imagination Technologies Ltd.
Gregoire Payen de La Garanderie

Intel Corporation
Jin Yang
Andy Chen
Josh Triplett
Sudarsana Nagineni
Jiajia Qin
Jiawei Shao
Jie Chen
Qiankun Miao
Bryan Bernhart
Yunchao He
Xinghua Cao
Brandon Jones

Klarlvdalens Datakonsult AB
Milian Wolff

Mozilla Corp.
Ehsan Akhgari
Edwin Flores
Jeff Gilbert
Mike Hommey
Benoit Jacob
Makoto Kato
Vladimir Vukicevic

Turbulenz
Michael Braithwaite

Ulrik Persson (ddefrostt)
Mark Banner (standard8mbp)
David Kilzer
Jacek Caban
Tibor den Ouden
Rgis Fnon
Sebastian Bergstein
James Ross-Gowan
Andrei Volykhin
Jrme Duval

Microsoft Corporation
Cooper Partin
Austin Kinross
Minmin Gong
Shawn Hargreaves

Microsoft Open Technologies, Inc.
Cooper Partin
Austin Kinross
NVIDIA Corporation
Olli Etuaho
Arun Patole
Qingqing Deng
Kimmo Kinnunen
Sami Visnen
Martin Radev

Opera Software ASA
Daniel Bratell
Tomasz Moniuszko
David Landell

Advanced Micro Devices, Inc.
Russ Lind
The MIT License (MIT)

Copyright (c) 2014 Arnout Kazemier

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2013 Linux Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials
provided with the distribution.

3. Neither the name of the Linux Foundation nor the names of its
contributors may be used to endorse or promote products
derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This BSD License conforms to the Open Source Initiative "Simplified
BSD License" as published at:
http://www.opensource.org/licenses/bsd-license.php
License information regarding the data obtained from National Land Survey of
Finland http://www.maanmittauslaitos.fi/en
- topographic model from Elevation model 2 m (U4421B, U4421D, U4422A and
  U4422C) 08/2014
- map image extracted from Topographic map raster 1:50 000 (U442) 08/2014

National Land Survey open data licence - version 1.0 - 1 May 2012

1. General information

The National Land Survey of Finland (hereinafter the Licensor), as the holder
of the immaterial rights to the data, has granted on the terms mentioned below
the right to use a copy (hereinafter data or dataset(s)) of the data (or a part
of it).

The Licensee is a natural or legal person who makes use of the data covered by
this licence. The Licensee accepts the terms of this licence by receiving the
dataset(s) covered by the licence.

This Licence agreement does not create a co-operation or business relationship
between the Licensee and the Licensor.
2. Terms of the licence

2.1. Right of use

This licence grants a worldwide, free of charge and irrevocable parallel right of use to open data. According to the terms of the licence, data received by the Licensee can be freely:
- copied, distributed and published,
- modified and utilised commercially and non-commercially,
- inserted into other products and
- used as a part of a software application or service.

2.2. Duties and responsibilities of the Licensee

Through reasonable means suitable to the distribution medium or method which is used in conjunction with a product containing data or a service utilising data covered by this licence or while distributing data, the Licensee shall:
- mention the name of the Licensor, the name of the dataset(s) and the time when the National Land Survey has delivered the dataset(s) (e.g.: contains data from the National Land Survey of Finland Topographic Database 06/2012)
- provide a copy of this licence or a link to it, as well as
- require third parties to provide the same information when granting rights to copies of dataset(s) or products and services containing such data and
- remove the name of the Licensor from the product or service, if required to do so by the Licensor.

The terms of this licence do not allow the Licensee to state in conjunction with the use of dataset(s) that the Licensor supports or recommends such use.

2.3. Duties and responsibilities of the Licensor

The Licensor shall ensure that
- the Licensor has the right to grant rights to the dataset(s) in accordance with this licence.

The data has been licensed "as is" and the Licensor
- shall not be held responsible for any errors or omissions in the data,
  disclaims any warranty for the validity or up to date status of the data and
  shall be free from liability for direct or consequential damages arising from the use of data provided by the Licensor,
- and is not obligated to ensure the continuous availability of the data, nor to announce in advance the interruption or cessation of availability, and the Licensor shall be free from liability for direct or consequential damages arising from any such interruption or cessation.

3. Jurisdiction

Finnish law shall apply to this licence.
4. Changes to this licence

The Licensor may at any time change the terms of the licence or apply a
different licence to the data. The terms of this licence shall, however, still
apply to such data that has been received prior to the change of the terms of
the licence or the licence itself.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,  
publicly display, publicly perform, sublicense, and distribute the  
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of  
this License, each Contributor hereby grants to You a perpetual,  
worldwide, non-exclusive, no-charge, royalty-free, irrevocable  
(except as stated in this section) patent license to make, have made,  
use, offer to sell, sell, import, and otherwise transfer the Work,  
where such license applies only to those patent claims licensable  
by such Contributor that are necessarily infringed by their  
Contribution(s) alone or by combination of their Contribution(s)  
with the Work to which such Contribution(s) was submitted. If You  
institute patent litigation against any entity (including a  
cross-claim or counterclaim in a lawsuit) alleging that the Work  
or a Contribution incorporated within the Work constitutes direct  
or contributory patent infringement, then any patent licenses  
granted to You under this License for that Work shall terminate  
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the  
Work or Derivative Works thereof in any medium, with or without  
modifications, and in Source or Object form, provided that You  
meet the following conditions:

(a) You must give any other recipients of the Work or  
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices  
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works  
    that You distribute, all copyright, patent, trademark, and  
    attribution notices from the Source form of the Work,  
    excluding those notices that do not pertain to any part of  
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its  
    distribution, then any Derivative Works that You distribute must  
    include a readable copy of the attribution notices contained  
    within such NOTICE file, excluding those notices that do not  
    pertain to any part of the Derivative Works, in at least one  
    of the following places: within a NOTICE text file distributed  
    as part of the Derivative Works; within the Source form or  
    documentation, if provided along with the Derivative Works; or,  
    within a display generated by the Derivative Works, if and  
    wherever such third-party notices normally appear. The contents  
    of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright 2011 The LibYuv Project Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

* Neither the name of Google nor the names of its contributors may
be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright 2014 PDFium Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions.
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, INCIDENTAL, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
(The MIT License)

Copyright (C) 2014-2017 by Vitaly Puzrin and Andrei Tuputcyn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.

// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be
// found in the LICENSE file.

// This file has been auto-generated from the Jinja2 template
// {{jinja_template_filename}}
// by the script {{code_generator}}.
// DO NOT MODIFY!

// clang-format off
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2011, Daniel Stenberg, <daniel@haxx.se>.
All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

The MIT License (MIT)

Copyright (c) 2015 Daniel Cousens

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Bodymovin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# License headers for subpackages

Transformation Matrix v2.0
(c) Epistemex 2014-2015
www.epistemex.com
By Ken Fyrstenberg
Contributions by leeoniya.
License: MIT, header required.
Copyright 2014 David Bau.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

BezierEasing - use bezier curve for transition easing function
by Gatan Renaudeau 2014 - 2015  MIT License

Credits: is based on Firefox's nsSMILKeySpline.cpp

Usage:
var spline = BezierEasing([ 0.25, 0.1, 0.25, 1.0 ])
spline.get(x) => returns the easing value | x must be in [0, 1] range

Copyright (c) 2011 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Internet Society, IETF or IETF Trust, nor
the names of specific contributors, may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2015-2019 The Khronos Group Inc.
Copyright (c) 2015-2019 LunarG, Inc.
Copyright (c) 2015-2019 Valve Corporation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on as "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Unless otherwise noted in the LICENSE.txt file in the install folder, all components of the Vulkan Runtime are licensed under the above license. Licenses for any components not so licensed are listed in the LICENSE.txt file.

==================================================================================

Copyright (c) 2009 Dave Gamble
Copyright (c) 2015-2016 The Khronos Group Inc.
Copyright (c) 2015-2016 Valve Corporation
Copyright (c) 2015-2016 LunarG, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

==================================================================================
=======================================================
Copyright (c) 2014 joseph werle <joseph.werle@gmail.com>
Copyright (c) 2015-2016 The Khronos Group Inc.
Copyright (c) 2015-2016 Valve Corporation
Copyright (c) 2015-2016 LunarG, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and/or associated documentation files (the "Materials"), to
deal in the Materials without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Materials, and to permit persons to whom the Materials are
furnished to do so, subject to the following conditions:

The above copyright notice(s) and this permission notice shall be included in
all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE
USE OR OTHER DEALINGS IN THE MATERIALS.

Copyright (c) 2014, Hugh Kennedy
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

---
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2010, The WebM Project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google, nor the WebM Project, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/*
 * Copyright (c) 2008 NVIDIA, Corporation
 */
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
* copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice (including the next
* paragraph) shall be included in all copies or substantial portions of the
* Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

KWSys - Kitware System Library
Copyright 2000-2009 Kitware, Inc., Insight Software Consortium
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the names of Kitware, Inc., the Insight Software Consortium,
nor the names of their contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010 by Armin Ronacher and contributors. See AUTHORS for more details.

Some rights reserved.

Redistribution and use in source and binary forms of the software as well as documentation, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The names of the contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE AND DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CMake - Cross Platform Makefile Generator
Copyright 2000-2018 Kitware, Inc. and Contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of Kitware, Inc. nor the names of Contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following individuals and institutions are among the Contributors:

* Aaron C. Meadows <cmake@shadowguarddev.com>
* Adriaan de Groot <groot@kde.org>
* Aleksey Avdeev <solo@altlinux.ru>
* Alexander Neundorf <neundorf@kde.org>
* Alexander Smorkalov <alexander.smorkalov@itseez.com>
* Alexey Sokolov <sokolov@google.com>
* Alex Turbov <i.zaufi@gmail.com>
* Andreas Pakulat <apaku@gmx.de>
* Andreas Schneider <asn@cryptomilk.org>
* Andr Rigland Brodtkorb <Andre.Brodtkorb@ifi.uio.no>
* Axel Huebl, Helmholtz-Zentrum Dresden - Rossendorf
* Benjamin Eikel
* Bjoern Ricks <bjoern.ricks@gmail.com>
* Brad Hards <bradh@kde.org>
* Christopher Harvey
* Christoph Gruening <foss@grueninger.de>
* Clement Creusot <creusot@cs.york.ac.uk>
* Daniel Blezak <blezak@gmail.com>
* Daniel Pfeifer <daniel@pfeifer-mail.de>
* Enrico Scholz <enrico.scholz@informatik.tu-chemnitz.de>
* Eran Ifrah <eran.ifrah@gmail.com>
* Esben Mose Hansen, Ange Optimization ApS
* Geoffrey Viola <geoffrey.viola@asirobots.com>
* Google Inc
* Gregor Jasny
* Helio Chissini de Castro <helio@kde.org>
* Ilya Lavrenov <ilya.lavrenov@itseez.com>
* Insight Software Consortium <insightsoftwareconsortium.org>
* Jan Woetzel
* Kelly Thompson <kgt@lanl.gov>
* Konstantin Podsvirov <konstantin@podsvirov.pro>
* Mario Bensi <mbensi@ipsquad.net>
* Mathieu Malatte <mathieu.malatte@gmail.com>
* Matthaeus G. Chajdas
* Matthias Kretz <kretz@kde.org>
* Matthias Maennich <matthias@maennich.net>
* Michael Strmer
* Miguel A. Figueroa-Villanueva
* Mike Jackson
* Mike McQuaid <mike@mikemcquaid.com>
* Nicolas Bock <nicolasbock@gmail.com>
* Nicolas Despres <nicolas.despres@gmail.com>
* Nikita Krunep'ko <krnekit@gmail.com>
* NVIDIA Corporation <www.nvidia.com>
* OpenGamma Ltd. <opengamma.com>
* Patrick Stotko <stotko@cs.uni-bonn.de>
* Per yvind Karlsen <peroyvind@mandriva.org>
* Peter Collingbourne <peter@pcc.me.uk>
* Petr Gotthard <gotthard@honeywell.com>
* Philip Lowman <philip@yhbt.com>
* Philippe Proulx <pproulx@efficios.com>
* Raffi Enficiaud, Max Planck Society
* Raumfeld <raumfeld.com>
* Roger Leigh <rleigh@codelibre.net>
* Rolf Eike Beer <eike@sf-mail.de>
* Roman Donchenko <roman.donchenko@itseez.com>
* Roman Kharitonov <roman.kharitonov@itseez.com>
* Ruslan Baratov
* Sebastian Holtermann <sebholt@xwmw.org>
* Stephen Kelly <steveire@gmail.com>
* Sylvain Joubert <joubert.sy@gmail.com>
* Thomas Sondergaard <ts@medical-insight.com>
* Tobias Hunger <tobias.hunger@qt.io>
* Todd Gamblin <tgamblin@llnl.gov>
* Tristan Carel
* University of Dundee
* Vadim Zhukov
* Will Dicharry <wdicharry@stellarscience.com>

See version control history for details of individual contributions.

The above copyright and license notice applies to distributions of CMake in source and binary form. Third-party software packages supplied with CMake under compatible licenses provide their own copyright notices.
documented in corresponding subdirectories or source files.

---------------------------------------------------------------------

CMake was initially developed by Kitware with the following sponsorship:

* National Library of Medicine at the National Institutes of Health
  as part of the Insight Segmentation and Registration Toolkit (ITK).

* US National Labs (Los Alamos, Livermore, Sandia) ASC Parallel
  Visualization Initiative.

* National Alliance for Medical Image Computing (NAMIC) is funded by the
  National Institutes of Health through the NIH Roadmap for Medical Research,
  Grant U54 EB005149.

* Kitware, Inc.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2009, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright
notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not
be used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization of the copyright holder.

MIT License

Copyright (c) 2014 Petka Antonov
          2015 Sindre Sorhus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of
   conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list
   of conditions and the following disclaimer in the documentation and/or other materials
   provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER(S) AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE COPYRIGHT HOLDER(S) OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Note: The license for SPIRV-Tools changed to Apache 2.0 via this commit:
https://github.com/KhronosGroup/SPIRV-Tools/commit/9fc8658ef301b0f03b2173d274c52f011b5c73e5

// Copyright (c) 2015-2016 The Khronos Group Inc.
// Copyright (c) 2015-2016 Google Inc.
// Copyright (c) 2016 LunarG Inc.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed...
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

**************************************************************************
** Copyright (C) YYYY The Qt Company Ltd.
** Contact: https://www.qt.io/licensing/
**
** This file is part of the FOO module of the Qt Toolkit.
**
** $QT_BEGIN_LICENSE:GPL-EXCEPT$
** Commercial License Usage
** Licensees holding valid commercial Qt licenses may use this file in
** accordance with the commercial license agreement provided with the
** Software or, alternatively, in accordance with the terms contained in
** a written agreement between you and The Qt Company. For licensing terms
** and conditions see https://www.qt.io/terms-conditions. For further
** information use the contact form at https://www.qt.io/contact-us.
**
** $QT_END_LICENSE$
**
** GNU General Public License Usage
** Alternatively, this file may be used under the terms of the GNU
** General Public License version 3 as published by the Free Software
** Foundation with exceptions as appearing in the file LICENSE.GPL3-EXCEPT
** included in the packaging of this file. Please review the following
** information to ensure the GNU General Public License requirements will
** be met: https://www.gnu.org/licenses/gpl-3.0.html.
**
** $QT_END_LICENSE$
**
****************************************************************************/

The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
(Copied from the README.)

LICENSE extracted from IJG's jpeg distribution:

--------------------------------------------------------------------------------

(Copied from the README.)
In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane. All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

GNU GENERAL PUBLIC LICENSE
The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.  
Contact: http://www.qt.io/licensing/  

You may use, distribute and copy the Qt Toolkit under the terms of GNU General Public License version 2, which is displayed below.

-------------------------------------------------------------------------
GNU GENERAL PUBLIC LICENSE  
Version 2, June 1991  

Copyright (C) 1989, 1991 Free Software Foundation, Inc.  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA  

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble  

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS
MIT License

Copyright (c) 2018 Nikita Skovoroda <chalkerx@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2000-2009  Josh Coalson
Copyright (C) 2011-2014  Xiph.Org Foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ISC License (ISC)
Copyright 2018 Stefan Penner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Copyright 1996 John Maloney and Mario Wolczko
Copyright 2003-2005 Tom Wu
Copyright 2005-2007 Sam Stephenson
Copyright 2006-2010 the V8 project authors

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

raytrace.js contains code from the the Prototype JavaScript framework, version 1.5.0:

Prototype JavaScript framework, version 1.5.0
Copyright (c) 2005-2007 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy of this
software and associated documentation files (the "Software"), to deal in the Software
without restriction, including without limitation the rights to use, copy, modify, merge,
publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
The majority of files in this project use the Apache 2.0 License. There are a few exceptions and their license can be found in the source. Any license deviations from Apache 2.0 are "more permissive" licenses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

Open Source Used In DNAC Platform Arica  6373
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

## -*-makefile-*-
## Copyright (C) 2016 and later: Unicode, Inc. and others.
## License & terms of use: http://www.unicode.org/copyright.html
## BSD-specific setup (FreeBSD, OpenBSD, NetBSD, *BSD)
## Copyright (c) 1999-2013, International Business Machines Corporation and
## others. All Rights Reserved.

## Commands to generate dependency files
GEN_DEPS.c=$$(CC) -E -MM $$(DEFS) $$(CPPFLAGS)
GEN_DEPS.cc=$$(CXX) -E -MM $$(DEFS) $$(CPPFLAGS)

## Flags for position independent code
SHAREDLIBCFLAGS = -fPIC
SHAREDLIBCXXFLAGS = -fPIC
SHAREDLIBCPPFLAGS = -DPIC

## Additional flags when building libraries and with threads
THREADSCPPFLAGS = -D_REENTRANT
LIBCPPFLAGS =

## Compiler switch to embed a runtime search path
LD_RPATH=
LD_RPATH_PRE=-Wl,-rpath,

## Compiler switch to embed a library name
LD_SONAME = -Wl,-soname -Wl,$(notdir $(MIDDLE_SO_TARGET))

## Shared library options
LD_SOOPTIONS= -Wl,-Bsymbolic
## Shared object suffix
SO = so

## Non-shared intermediate object suffix
STATIC_O = ao

## Compilation rules
%.$(STATIC_O): $(srcdir)/%.c
$(COMPILE.c) $(STATICCPPFLAGS) $(STATICCFLAGS) -o $@ $<
%.o: $(srcdir)/%.c
$(COMPILE.c) $(DYNAMICCPPFLAGS) $(DYNAMICCFLAGS) -o $@ $<
%.$(STATIC_O): $(srcdir)/%.cpp
$(COMPILE.cc) $(STATICCPPFLAGS) $(STATICCXXFLAGS) -o $@ $<
%.o: $(srcdir)/%.cpp
$(COMPILE.cc) $(DYNAMICCPPFLAGS) $(DYNAMICCXXFLAGS) -o $@ $<

## Dependency rules
%.d: $(srcdir)/%.c
@echo "generating dependency information for $<"
@$SHELL -ec '$(GEN_DEPS.c) $< \
| sed "s/\(\*\)/\1.o $@ : /g\" > $@; \
[ -s $@ ] || rm -f $@'
%.d: $(srcdir)/%.cpp
@echo "generating dependency information for $<"
@$SHELL -ec '$(GEN_DEPS.cc) $< \
| sed "s/\(\*\)/\1.o $@ : /g\" > $@; \
[ -s $@ ] || rm -f $@'

## Versioned libraries rules
%.$(SO).$(SO_TARGET_VERSION_MAJOR): %.$(SO).$(SO_TARGET_VERSION)
$(RM) $@ && ln -s ${<F} $@
%.$(SO): %.$(SO).$(SO_TARGET_VERSION_MAJOR)
$(RM) $@ && ln -s ${*F}.$(SO).$(SO_TARGET_VERSION) $@

## Bind internal references

# LDflags that pkgdata will use
BIR_LDFLAGS= -Wl,-Bsymbolic

# Dependencies [i.e. map files] for the final library
BIR_DEPS=

## Remove shared library 's'
STATIC_PREFIX_WHEN_USED =
STATIC_PREFIX =
BSD License https://opensource.org/licenses/BSD-3-Clause

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holders, nor those of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
MIT License

Copyright (c) 2017 Intel Corporation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// Copyright (c) 2012 The Polymer Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
The source code of UnRAR utility is freeware. This means:

1. All copyrights to RAR and the utility UnRAR are exclusively owned by the author - Alexander Roshal.

2. UnRAR source code may be used in any software to handle RAR archives without limitations free of charge, but cannot be used to develop RAR (WinRAR) compatible archiver and to re-create RAR compression algorithm, which is proprietary. Distribution of modified UnRAR source code in separate form or as a part of other software is permitted, provided that full text of this paragraph, starting from "UnRAR source code" words, is included in license, or in documentation if license is not available, and in source code comments of resulting package.

3. The UnRAR utility may be freely distributed. It is allowed to distribute UnRAR inside of other software packages.

4. THE RAR ARCHIVER AND THE UnRAR UTILITY ARE DISTRIBUTED "AS IS". NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED. YOU USE AT YOUR OWN RISK. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS, DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING OR MISUSING THIS SOFTWARE.

5. Installing and using the UnRAR utility signifies acceptance of these terms and conditions of the license.

6. If you don't agree with terms of the license you must remove UnRAR files from your storage devices and cease to use the utility.

Thank you for your interest in RAR and UnRAR.

Alexander L. Roshal
// Copyright 2014 The Chromium OS Authors. All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
# The MIT License (MIT)

**Copyright (c) Rod Vagg (the "Original Author") and additional contributors**

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
// Copyright (c) 2005, Google Inc.
// All rights reserved.
//
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2000 Ming-Che Chuang
Copyright (C) 2001, 2002 James Su, Turbolinux Inc.
Copyright (C) 2002 WU Yi, HancomLinux Inc.
Copyright (C) 2001, 2002 Anthony Fok, ThizLinux Laboratory Ltd.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
MIT License

Copyright (c) DC <threedeecee@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2012 Irakli Gozalishvili. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU Lesser General Public License version 3. That license references the General Public License version 3, that is displayed below. Other portions of the Qt Toolkit may be licensed directly under this license.
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for
infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by the
Corresponding Source fixed on a durable physical medium
customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by a
written offer, valid for at least three years and valid for as
long as you offer spare parts or customer support for that product
model, to give anyone who possesses the object code either (1) a
copy of the Corresponding Source for all the software in the
product that is covered by this License, on a durable physical
medium customarily used for software interchange, for a price no
more than your reasonable cost of physically performing this
conveying of source, or (2) access to copy the
Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the
written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord
with subsection 6b.

d) Convey the object code by offering access from a designated
place (gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party)
that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the
Corresponding Source. Regardless of what server hosts the
Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims
owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is
in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published
by the Free Software Foundation.

If the Program specifies that a proxy can decide which future
versions of the GNU General Public License can be used, that proxy’s
public statement of acceptance of a version permanently authorizes you
to choose that version for the Program.

Later license versions may give you additional or different
permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a
later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

```
#!/usr/bin/env node

var nopt = require('./lib/nopt')
, path = require('path')

, types = { num: Number,
          , bool: Boolean
          , help: Boolean
          , list: Array
          , "num-list": [Number, Array]
          , "str-list": [String, Array]
          , "bool-list": [Boolean, Array]
          , str: String
          , clear: Boolean
          , config: Boolean
          , length: Number
          , file: path
          }

, shorthands = { s: [ "--str", "astring" ]
               , b: [ "--bool" ]
               , nb: [ "--no-bool" ]
               , tft: [ "--bool-list", "--no-bool-list", "--bool-list", "true" ]
               , ":": ["--help"]
               , h: ["--help"]
               , H: ["--help"]
               , n: [ "--num", "125" ]
               , c: ["--config"]
               , l: ["--length"]
               , f: ["--file"]
          }

, parsed = nopt( types
                , shorthands
                , process.argv
                , 2 )

console.log("parsed", parsed)

if (parsed.help) {
    console.log(""
    console.log("nopt cli tester")
    console.log(""
    console.log("types")
    console.log(Object.keys(types).map(function M (t) {
        var type = types[t]
        if (Array.isArray(type)) {
            
```
The "Clarified Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Distribution fee" is a fee you charge for providing a copy of this Package to another party.

"Freely Available" means that no fee is charged for the right to use the item, though there may be fees involved in handling the item.
1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain, or those made Freely Available, or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site allowing unrestricted access to them, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

   b) use the modified Package only within your corporation or organization.

   c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

   e) permit and encourage anyone who receives a copy of the modified Package permission to make your modifications Freely Available in some specific way.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

   a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

   b) accompany the distribution with the machine-readable source of the Package with your modifications.

   c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.
d) make other distribution arrangements with the Copyright Holder.

e) offer the machine-readable source of the Package, with your modifications, by mail order.

5. You may charge a distribution fee for any distribution of this Package. If you offer support for this Package, you may charge any fee you choose for that support. You may not charge a license fee for the right to use this Package itself. You may distribute this Package in aggregate with other (possibly commercial and possibly nonfree) programs as part of a larger (possibly commercial and possibly nonfree) software distribution, and charge license fees for other parts of that software distribution, provided that you do not advertise this Package as a product of your own. If the Package includes an interpreter, You may embed this Package’s interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of the Standard Version of the Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
Open Source Used In DNAC Platform Arica 6400

IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End
(Copied from the COPYRIGHT file of https://code.google.com/p/sctp-refimpl/source/browse/trunk/COPYRIGHT)

--------------------------------------------------------------------------------

Copyright (c) 2001, 2002 Cisco Systems, Inc.
Copyright (c) 2002-12 Randall R. Stewart
Copyright (c) 2002-12 Michael Tuexen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

C++ Big Integer Library
(see ChangeLog for version)

http://mattmccutchen.net/bigint/

Written and maintained by Matt McCutchen <matt@mattmccutchen.net>

You can use this library in a C++ program to do arithmetic on integers of size limited only by your computer's memory. The library provides BigUnsigned and BigInteger classes that represent nonnegative integers and signed integers, respectively. Most of the C++ arithmetic operators are overloaded for these classes, so big-integer calculations are as easy as:
#include "BigIntegerLibrary.hh"

BigInteger a = 65536;
cout << (a * a * a * a * a * a * a * a);

(prints 340282366920938463463374607431768211456)

The code in `sample.cc' demonstrates the most important features of the library. To get started quickly, read the code and explanations in that file and run it. If you want more detail or a feature not shown in `sample.cc', consult the actual header and source files, which are thoroughly commented.

This library emphasizes ease of use and clarity of implementation over speed; some users will prefer GMP (http://swox.com/gmp/), which is faster. The code is intended to be reasonably portable across computers and modern C++ compilers; in particular, it uses whatever word size the computer provides (32-bit, 64-bit, or otherwise).

Compiling programs that use the library
---------------------------------------
The library consists of a folder full of C++ header files (`.hh') and source files (`.cc'). Your own programs should `#include' the necessary header files and link with the source files. A makefile that builds the sample program (`sample.cc') is included; you can adapt it to replace the sample with your own program.

Alternatively, you can use your own build system or IDE. In that case, you must put the library header files where the compiler will find them and arrange to have your program linked with the library source files; otherwise, you will get errors about missing header files or "undefined references". To learn how to do this, consult the documentation for the build system or IDE; don't bother asking me. Adding all the library files to your project will work in many IDEs but may not be the most desirable approach.

Resources
---------
The library's Web site (above) provides links to released versions, the current development version, and a mailing list for release announcements, questions, bug reports, and other discussion of the library. I would be delighted to hear from you if you like this library and/or find a good use for it.

Bugs and enhancements
---------------------
The library has been tested by me and others but is by no means bug-free. If you find a bug, please report it, whether it comes in the form of compiling trouble, a mathematically inaccurate result, or a memory-management blooper (since I use Java, these are altogether too common in my C++). I generally fix all reported bugs. You are also welcome to request enhancements, but I am
unlikely to do substantial amounts of work on enhancements at this point.

Legal
-----
I, Matt McCutchen, the sole author of the original Big Integer Library, waive my
copyright to it, placing it in the public domain. The library comes with
absolutely no warranty.

~~~
The MIT License (MIT)

Copyright (c) 2014-2017 createECDH contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License

Open Source Used In DNAC Platform Arica 6408
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==============================================================================
Gradle Subcomponents:

License for the slf4j package

SLF4J License

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

These terms are identical to those of the MIT License, also called the X License or the X11 License, which is a simple, permissive non-copyleft free software license. It is deemed compatible with virtually all types of licenses, commercial or otherwise. In particular, the Free Software Foundation has declared it compatible with GNU GPL. It is also known to be approved by the Apache Software Foundation as compatible with Apache Software License.

License for the JUnit package

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections
2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

License for the JCIFS package

JCIFS License

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.
We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the
Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a
portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is
included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for
making modifications to it. For a library, complete source code means
all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation
and installation of the library.

Activities other than copying, distribution and modification are not
covered by this License: they are outside its scope. The act of
running a program using the Library is not restricted, and output from
such a program is covered only if its contents constitute a work based
on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that
you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact
all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the
Library.

You may charge a fee for the physical act of transferring a copy,
and you may at your option offer warranty protection in exchange for a
fee.

2. You may modify your copy or copies of the Library or any portion
of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices
      stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no
      charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a
      table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or
link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice

That's all there is to it!

BSD License

Copyright (c) 2000-2015 www.hamcrest.org  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.
Copyright (C) 2011-2017 by Yehuda Katz

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

# Files that don't get a copyright, or which are taken from elsewhere.
#
# All lines in this file are patterns, including the comment lines; this
# means that e.g. `FTL.TXT' matches all files that have this string in
# the file name (including the path relative to the current directory,
# always starting with `./').
#
# Don't put empty lines into this file!
#
.gitignore
#
built/unix/pkg.m4
#
docs/FTL.TXT
docs/GPLv2.TXT
#
include/freetype/internal/fthash.h
#
src/base/fthash.c
src/base/md5.c
src/base/md5.h
#
src/bdf/bdf.c
src/bdf/bdf.h
src/bdf/bdfdrivr.c
src/bdf/bdfdrivr.h
src/bdf/bdferror.h
src/bdf/bdflib.c
src/bdf/module.mk
src/bdf/README
src/bdf/rules.mk
#
src/pcf/module.mk
src/pcf/pcf.c
src/pcf/pcf.h
src/pcf/pcfdrivr.c
src/pcf/pcfdrivr.h
src/pcf/pcferror.h
src/pcf/pcfread.c
src/pcf/pcfread.h
The MIT License (MIT)

Copyright (c) 2015 Gal Koren

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License
Original Library
- Copyright (c) Marak Squires

Additional Functionality
- Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Fonts are (c) Bitstream (see below). DejaVu changes are in public domain. Glyphs imported from Arev fonts are (c) Tavmjong Bah (see below)

Bitstream Vera Fonts Copyright
-------------------------------
Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Bitstream" or the word
This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Bitstream Vera" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of Gnome, the Gnome Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the Gnome Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

Arev Fonts Copyright
-----------------------

Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the modifications to the Bitstream Vera Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either
the words "Tavmjong Bah" or the word "Arev".

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Tavmjong Bah Arev" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL TAVMJONG BAH BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the name of Tavmjong Bah shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from Tavmjong Bah. For further information, contact: tavmjong @ free . fr.

TeX Gyre DJV Math
-----------------
Fonts are (c) Bitstream (see below). DejaVu changes are in public domain.

Math extensions done by B. Jackowski, P. Strzelczyk and P. Pianowski (on behalf of TeX users groups) are in public domain.

Letters imported from Euler Fraktur from AMSfonts are (c) American Mathematical Society (see below).

Bitstream Vera Fonts Copyright
Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license (Fonts) and associated documentation files (the Font Software), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:
The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words Bitstream or the word Vera.

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the Bitstream Vera names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE. Except as contained in this notice, the names of GNOME, the GNOME Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the GNOME Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

AMSFonts (v. 2.2) copyright

The PostScript Type 1 implementation of the AMSFonts produced by and previously distributed by Blue Sky Research and Y&Y, Inc. are now freely available for general use. This has been accomplished through the cooperation of a consortium of scientific publishers with Blue Sky Research and Y&Y. Members of this consortium include:
In order to assure the authenticity of these fonts, copyright will be held by the American Mathematical Society. This is not meant to restrict in any way the legitimate use of the fonts, such as (but not limited to) electronic distribution of documents containing these fonts, inclusion of these fonts into other public domain or commercial font collections or computer applications, use of the outline data to create derivative fonts and/or faces, etc. However, the AMS does require that the AMS copyright notice be removed from any derivative versions of the fonts which have been altered in any way. In addition, to ensure the fidelity of TeX documents using Computer Modern fonts, Professor Donald Knuth, creator of the Computer Modern faces, has requested that any alterations which yield different font metrics be given a different name.

// sigslot.h: Signal/Slot classes

// Written by Sarah Thompson (sarah@telergy.com) 2002.

// License: Public domain. You are free to use this code however you like, with the proviso that the author takes on no responsibility or liability for any use.

Copyright (c) 2006, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2011, Daniel Stenberg, <daniel@haxx.se>.
All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Copyright (c) 1999 Apple Computer, Inc. All rights reserved.

@APPLE_LICENSE_HEADER_START@

This file contains Original Code and/or Modifications of Original Code as defined in and that are subject to the Apple Public Source License Version 2.0 (the 'License'). You may not use this file except in compliance with the License. Please obtain a copy of the License at http://www.opensource.apple.com/apsl/ and read it before using this file.

The Original Code and all software distributed under the License are distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT. Please see the License for the specific language governing rights and limitations under the License.
Copyright 2007-2008 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

(MIT)

Copyright (c) 2013 Julian Gruber &lt;julian@juliangruber.com&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"). to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

All files in this directory are provided by the following license if not stated otherwise in the individual file:

====================
This file is provided under a dual BSD/GPLv2 license. When using or redistributing this file, you may do so under either license.

GPL LICENSE SUMMARY

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.
This program is free software; you can redistribute it and/or modify it under the terms of version 2 of the GNU General Public License as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St - Fifth Floor, Boston, MA 02110-1301 USA. The full GNU General Public License is included in this distribution in the file called LICENSE.GPL.

Contact Information:

BSD LICENSE

Copyright (c) 2005-2012 Intel Corporation. All rights reserved.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 1985, 1986 The Regents of the University of California.  
All rights reserved.

This code is derived from software contributed to Berkeley by 
James A. Woods, derived from original work by Spencer Thomas 
and Joseph Orost.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions 
are met:
1. Redistributions of source code must retain the above copyright 
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright 
   notice, this list of conditions and the following disclaimer in the 
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software 
   must display the following acknowledgement: 
   This product includes software developed by the University of 
   California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors 
   may be used to endorse or promote products derived from this software 
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND 
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE 
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL 
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS 
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) 
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT 
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY 
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF 
SUCH DAMAGE.

ISC License

Copyright (c) 2017, Mapbox

Permission to use, copy, modify, and/or distribute this software for any purpose 
with or without fee is hereby granted, provided that the above copyright notice 
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH 
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND 
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, 
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM 
LOSS

Open Source Used In DNAC Platform Arica 6441
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.
JSZip is dual licensed. You may use it under the MIT license *or* the GPLv3
license.

The MIT License
===============

Copyright (c) 2009-2012 Stuart Knightley, David Duponchel, Franz Buchinger, Antonio Afonso

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

GPL version 3
============

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a
You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.
5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the
additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.
Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a
party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

NASM is now licensed under the 2-clause BSD license, also known as the simplified BSD license.

Copyright 1996-2010 the NASM Authors - All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

cubemaps:  See cubemaps/miramar/README.TXT

gltf:      See gltf/LICENSE.md

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

The MIT License (MIT)

Copyright (c) 2015-2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Kevin Roark

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
The Qt Toolkit is Copyright (C) 2015 The Qt Company Ltd.
Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below.

-------------------------------------------------------------------------
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this
licensedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, this License refers to version 3 of the GNU Lesser
General Public License, and the GNU GPL refers to version 3 of the
GNU General Public License.

The Library refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An Application is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A Combined Work is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the Linked
Version.

The Minimal Corresponding Source for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The Corresponding Application Code for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.
1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

   a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

   b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

   a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

   b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

   a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License or any later version applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Copyright 2006-2011, the V8 project authors. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving
lengthy legal documents to sign. The sources are provided for free but without
warranty of any kind. The library has been entirely written by Jean-loup
Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in
the file ChangeLog history information documenting your changes. Please read
the FAQ for more information on the distribution of modified source versions.
The MIT License (MIT)

Copyright (c) 2014-2017 browserify-aes contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

Open Source Used In DNAC Platform Arica  6462
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2010
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Mocking Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Benoit Sigoure <tsuna@google.com>
Bogdan Piloca <boo@google.com>
Chandler Carruth <chandlerc@google.com>
Dave MacLachlan <dmaclach@gmail.com>
David Anderson <danderson@google.com>
Dean Sturtevant
Gene Volovich <gv@cite.com>
Hal Burch <gmock@hburch.com>
This is the GNU General Public License version 3, annotated with The Qt Company GPL Exception 1.0:

The Qt Company GPL Exception 1.0

Exception 1:

As a special exception you may create a larger work which contains the output of this application and distribute that work under terms of your choice, so long as the work is not otherwise derived from or based on this application and so long as the work does not in itself generate output that contains the output from this application in its original or modified form.

Exception 2:

As a special exception, you have permission to combine this application with Plugins licensed under the terms of your choice, to produce an executable, and to copy and distribute the resulting executable under
the terms of your choice. However, the executable must be accompanied by a prominent notice offering all users of the executable the entire source code to this application, excluding the source code of the independent modules, but including any changes you have made to this application, under the terms of this license.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS**

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.
A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive
interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

   a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

   b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

   c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

   d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).
The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the
violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.
A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.
A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".
You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>. Copyright (c) 1998-2013, Brian Gladman, Worcester, UK. All rights reserved.

The redistribution and use of this software (with or without changes) is allowed without the payment of fees or royalties provided that:

source code distributions include the above copyright notice, this list of conditions and the following disclaimer;

binary distributions include the above copyright notice, this list of conditions and the following disclaimer in their documentation.

This software is provided 'as is' with no explicit or implied warranties in respect of its operation, including, but not limited to, correctness and fitness for purpose.

GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2017 The Qt Company Ltd. Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below. This license makes reference to the version 3 of the GNU General Public License, which you can find in the LICENSE.GPLv3 file.

-------------------------------------------------------------------------

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Everyone is permitted to copy and distribute verbatim copies of this licensedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.
0. Additional Definitions.

As used herein, this License refers to version 3 of the GNU Lesser General Public License, and the GNU GPL refers to version 3 of the GNU General Public License.

The Library refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An Application is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A Combined Work is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the Linked Version.

The Minimal Corresponding Source for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The Corresponding Application Code for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of
this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

   a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

   b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

   a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

   b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

   c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

   d) Do one of the following:

      0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

      1) Use a suitable shared library mechanism for linking with
the Library. A suitable mechanism is one that (a) uses at run
time a copy of the Library already present on the user's
computer system, and (b) will operate properly with a modified
version of the Library that is interface-compatible with the
Linked Version.

e) Provide Installation Information, but only if you would
otherwise be required to provide such information under section 6
of the GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the
Combined Work produced by recombining or relinking the Application
with a modified version of the Linked Version. (If you use option
4d0, the Installation Information must accompany the Minimal
Corresponding Source and Corresponding Application Code. If you
use option 4d1, you must provide the Installation Information in
the manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.)


You may place library facilities that are a work based on the Library
side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of
it is a work based on the Library, and explaining where to find
the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
as you received it specifies that a certain numbered version of the
GNU Lesser General Public License or any later version applies to
it, you have the option of following the terms and conditions either
of that published version or of any later version published by the
Free Software Foundation. If the Library as you received it does not
specify a version number of the GNU Lesser General Public License,
you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2016 The Qt Company Ltd. Contact: http://www.qt.io/licensing/

You may use, distribute and copy the Qt GUI Toolkit under the terms of GNU Lesser General Public License version 3. That license references the General Public License version 3, that is displayed below. Other portions of the Qt Toolkit may be licensed directly under this license.

-------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run
modified versions of the software inside them, although the manufacturer
can do so. This is fundamentally incompatible with the aim of
protecting users' freedom to change the software. The systematic
pattern of such abuse occurs in the area of products for individuals to
use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those
products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions
of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.
States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could
make it effectively proprietary. To prevent this, the GPL assures that
patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and
modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of
works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major
Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.
Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the
written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord
with subsection 6b.

d) Convey the object code by offering access from a designated
place (gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party)
that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the
Corresponding Source. Regardless of what server hosts the
Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided
you inform other peers where the object code and Corresponding
Source of the work are being offered to the general public at no
charge under subsection 6d.

A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family,
or household purposes, or (2) anything designed or sold for incorporation
into a dwelling. In determining whether a product is a consumer product,
doubtful cases shall be resolved in favor of coverage. For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product. A product
is a consumer product regardless of whether the product has substantial
commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source. The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.
If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the
terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.
You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work
in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this
License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

# Rollup core license
Rollup is released under the MIT license:

The MIT License (MIT)
Copyright (c) 2017 [these people](https://github.com/rollup/rollup/graphs/contributors)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

# Licenses of bundled dependencies
The published Rollup artifact additionally contains code with the following licenses:
MIT, ISC, Apache-2.0

# Bundled dependencies:
## acorn-import-meta
License: MIT
By: Adrian Heine
Repository: https://github.com/adrianheine/acorn-import-meta

> Copyright (C) 2017-2018 by Adrian Heine
>
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------

## acorn-walk
License: MIT
By: Marijn Haverbeke, Ingvar Stepanyan, Adrian Heine
Repository: https://github.com/acornjs/acorn.git

> Copyright (C) 2012-2018 by various contributors (see AUTHORS)
>
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------

## braces
License: MIT
By: Jon Schlinkert, Brian Woodward, Elan Shanker, Eugene Sharygin, hemanth.hm
Repository: micromatch/braces

> The MIT License (MIT)
>
> Copyright (c) 2014-2018, Jon Schlinkert.
>
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------

## date-time
License: MIT
By: Sindre Sorhus
Repository: sindresorhus/date-time

---------------------------------------

## fill-range
License: MIT
By: Jon Schlinkert, Edo Rivai, Paul Miller, Rouven Weling
> The MIT License (MIT)
> 
> Copyright (c) 2014-present, Jon Schlinkert.
> 
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
> 
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
> 
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

## hash.js

License: MIT

By: Fedor Indutny

Repository: git@github.com:indutny/hash.js

## immutable

License: MIT

By: Lee Byron

Repository: git://github.com/facebook/immutable-js.git

> MIT License
> 
> Copyright (c) 2014-present, Facebook, Inc.
> 
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### inherits
License: ISC
Repository: git://github.com/isaacs/inherits

The ISC License

Copyright (c) Isaac Z. Schlueter

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHERWISE ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

### is-number
License: MIT
By: Jon Schlinkert, Olsten Larck, Rouven Weling
Repository: jonschlinkert/is-number

The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

## is-reference
License: MIT
By: Rich Harris
Repository: git+https://github.com/Rich-Harris/is-reference.git

## locate-character
License: MIT
By: Rich Harris
Repository: Rich-Harris/locate-character

## magic-string
License: MIT
By: Rich Harris
Repository: https://github.com/rich-harris/magic-string

Copyright 2018 Rich Harris

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---------------------------------------

## micromatch
License: MIT
By: Jon Schlinkert, Amila Welihinda, Bogdan Chadkin, Brian Woodward, Devon Govett, Elan Shanker, Fabrcio
Matt, Martin Kolrik, Olsten Larck, Paul Miller, Tom Byrer, Tyler Akins, Peter Bright
Repository: micromatch/micromatch

> The MIT License (MIT)

> Copyright (c) 2014-present, Jon Schlinkert.

> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:

> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.

> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------

## minimalistic-assert
License: ISC
Repository: https://github.com/calvinmetcalf/minimalistic-assert.git

> Copyright 2015 Calvin Metcalf

> Permission to use, copy, modify, and/or distribute this software for any purpose
> with or without fee is hereby granted, provided that the above copyright notice
> and this permission notice appear in all copies.

> | Open Source Used In DNAC Platform Arica 6507 |
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---------------------------------------

## minimist
License: MIT
By: James Halliday
Repository: git://github.com/substack/minimist.git

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---------------------------------------

## parse-ms
License: MIT
By: Sindre Sorhus
Repository: sindresorhus/parse-ms

---------------------------------------

## picomatch
License: MIT
By: Jon Schlinkert
Repository: micromatch/picomatch
> The MIT License (MIT)
>
> Copyright (c) 2017-present, Jon Schlinkert.
>
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

`-------------
## pretty-bytes
License: MIT
By: Sindre Sorhus
Repository: sindresorhus/pretty-bytes

-------------

## pretty-ms
License: MIT
By: Sindre Sorhus
Repository: sindresorhus/pretty-ms

-------------

## require-relative
License: MIT
By: Valerio Proietti
Repository: git://github.com/kamicane/require-relative.git

-------------

## rollup-pluginutils
License: MIT

-------------
By: Rich Harris
Repository: rollup/rollup-plugin-utils

---------------------------------------

## signal-exit
License: ISC
By: Ben Coe
Repository: https://github.com/tapjs/signal-exit.git

> The ISC License
>
> Copyright (c) 2015, Contributors
>
> Permission to use, copy, modify, and/or distribute this software
> for any purpose with or without fee is hereby granted, provided
> that the above copyright notice and this permission notice
> appear in all copies.
>
> THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
> WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
> OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE
> LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES
> OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
> WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
> ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---------------------------------------

## sourcemap-codec
License: MIT
By: Rich Harris
Repository: https://github.com/Rich-Harris/sourcemap-codec

> The MIT License
>
> Copyright (c) 2015 Rich Harris
>
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------
## time-zone
License: MIT
By: Sindre Sorhus
Repository: sindresorhus/time-zone

---------------------------------------
## to-regex-range
License: MIT
By: Jon Schlinkert, Rouven Weling
Repository: micromatch/to-regex-range

> The MIT License (MIT)
> Copyright (c) 2015-present, Jon Schlinkert.
> Permission is hereby granted, free of charge, to any person obtaining a copy
> of this software and associated documentation files (the "Software"), to deal
> in the Software without restriction, including without limitation the rights
> to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
> copies of the Software, and to permit persons to whom the Software is
> furnished to do so, subject to the following conditions:
> The above copyright notice and this permission notice shall be included in
> all copies or substantial portions of the Software.
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
> THE SOFTWARE.

---------------------------------------
## tslib
License: Apache-2.0
By: Microsoft Corp.
Repository: https://github.com/Microsoft/tslib.git

> Apache License
>
> Version 2.0, January 2004
>
> http://www.apache.org/licenses/
>
> TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
>
> 1. Definitions.
>
> "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
>
> "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
>
> "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
>
> "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
>
> "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
>
> "Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
>
> "Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
>
> "Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
>
> "Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensee for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

> "Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

> 2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

> 3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

> 4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

> You must give any other recipients of the Work or Derivative Works a copy of this License; and

> You must cause any modified files to carry prominent notices stating that You changed the files; and

> You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

> If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided alongside the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

> 5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without
any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

> 6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

> 7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

> 8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

> 9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

> END OF TERMS AND CONDITIONS

---------------------------------------

## turbocolor
License: MIT
By: Jorge Bucaran
Repository: jorgebucaran/turbocolor

> Copyright 2015-present [Jorge Bucaran](https://github.com/jorgebucaran)

> Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

> The above copyright notice and this permission notice shall be included in all
OPEN SOURCE USED IN DNAC PLATFORM ARTICA 6515

> copies or substantial portions of the Software.
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
> IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
> FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
> AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
> LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
> OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
> SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR

Open Source Used In DNAC Platform Artica 6515
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) MapBox
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name "MapBox" nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2007 Red Hat, inc

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The MIT License (MIT)

Copyright (c) Elan Shanker and Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.
A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If
you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Copyright 2019 "David Mark Clements <david.mark.clements@gmail.com>"

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation
The rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright Joyent, Inc. and other Node contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014-2017 Calvin Metcalf & contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012-2015, JP Richardson <jprichardson@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2011-2013 Jake Luer jake@alogicalparadox.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

thestig@chromium.org

Copyright (c) 2015, Valve Corporation

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

TLS Lite includes code from different sources. All code is either dedicated to the public domain by its authors, or available under a BSD-style license. In particular:

- Code written by Trevor Perrin, Kees Bos, Sam Rushing, Dimitris Moraitis, Marcelo Fernandez, Martin von Loewis, Dave Baggett, and Yngve Pettersen is available under the following terms:

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.
In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Code written by Bram Cohen (rijndael.py) was dedicated to the public domain by its author. See rijndael.py for details.

Code written by Google is available under the following terms:

Copyright (c) 2008, The Chromium Authors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2012-2016 Mickael Daniel

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (c) 2015 John Hiesey

Permission is hereby granted, free of charge,
without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The ISC License

Copyright (c) 2015 Elan Shanker

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The MIT License

Copyright (c) 2013 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright 2012-2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the
  names of its contributors may be used to endorse or promote products
  derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2017 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
The MIT License (MIT)

Copyright (c) 2017 Bart Muzzin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Derived from:
The MIT License (MIT)

Copyright (c) 2015 mojmir svoboda

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013, Ethan Furman.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name Ethan Furman nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NOTE! This copyright does *not* cover user programs that use kernel services by normal system calls - this is merely considered normal use of the kernel, and does *not* fall under the heading of "derived work". Also note that the GPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the linux kernel) is copyrighted by me and others who actually wrote it.

Linus Torvalds

----------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Copyright 2001-2011 Xiph.Org, Skype Limited, Octasic, Jean-Marc Valin, Timothy B. Terriberry, CSIRO, Gregory Maxwell, Mark Borgerding,
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Internet Society, IETF or IETF Trust, nor the names of specific contributors, may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Opus is subject to the royalty-free patent licenses which are specified at:

Xiph.Org Foundation:
https://datatracker.ietf.org/ipr/1524/

Microsoft Corporation:
https://datatracker.ietf.org/ipr/1914/

Broadcom Corporation:
https://datatracker.ietf.org/ipr/1526/
(Copied from the README.)

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you,
its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.  
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this  
software (or portions thereof) for any purpose, without fee, subject to these  
conditions:
(1) If any part of the source code for this software is distributed, then this  
README file must be included, with this copyright and no-warranty notice  
unaltered; and any additions, deletions, or changes to the original files  
must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying  
documentation must state that "this software is based in part on the work of  
the Independent JPEG Group".
(3) Permission for use of this software is granted only if the user accepts  
full responsibility for any undesirable consequences; the authors accept  
NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code,  
not just to the unmodified library. If you use our work, you ought to  
acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name  
in advertising or publicity relating to this software or products derived from  
it. This software may be referred to only as "the Independent JPEG Group's  
software".

We specifically permit and encourage the use of this software as the basis of  
commercial products, provided that all warranty or liability claims are  
assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch,  
sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.  
ansi2knr.c is NOT covered by the above copyright and conditions, but instead  
by the usual distribution terms of the Free Software Foundation; principally,  
that you must include source code if you redistribute it. (See the file  
ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part  
of any program generated from the IJG code, this does not limit you more than  
the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf.  
It is copyright by the Free Software Foundation but is freely distributable.  
The same holds for its supporting scripts (config.guess, config.sub,  
ltconfig, ltmain.sh). Another support script, install-sh, is copyright  
by M.I.T. but is also freely distributable.
It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.) So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that

"The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

jconfig.h is distributed under the MPL 1.1/GPL 2.0/LGPL 2.1 tri-license.

jmorecfg.h contains modifications, which are distributed under the Netscape Public License.
# Copyright (c) 2015 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.**

Copyright JS Foundation and other contributors, https://js.foundation
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# Names should be added to this file as:
# Name <email address>
Google Inc.
Nicolas Capens <capn@google.com>
Alexis Hu <sugoi@google.com>
Shannon Woods <shannonwoods@google.com>
Corentin Wallez <cwallez@google.com>
Greg Hartman <ghartman@google.com>
Ping-Hao Wu <pinghao@google.com>
Maxime Grgoire <mgregoire@google.com>
Veranika Liaukevich <veranika@google.com>
John Bauman <jbauman@google.com>
Keun Soo Yim <yim@google.com>
John Sheu <sheu@google.com>
Philippe Hamel <hamelphi@google.com>
Daniel Toyama <kenjitoyama@google.com>
Meng-Lin Wu <marleymoo@google.com>
Krzysztof Kosiski <krzystio@google.com>
Chris Forbes <chrisforbes@google.com>
Ben Clayton <bclayton@google.com>
Hernan Liatis <hliatis@google.com>
Logan (Tzu-hsiang) Chien <loganchien@google.com>
Stephen White <senorblanco@google.com>
Raymond Chiu <chiur@google.com>

TransGaming Inc.
Nicolas Capens
Gavriel State
Jim MacArthur
Daniel Koch
Luther Johnson
Rob Stepinski
(The MIT License)

Copyright (c) 2017, Ryan Zimmerman <opensrc@ryanzim.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the 'Software'), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. This license is for the example glTF code, not the [specification](specification/README.md) and [schema](specification).

Original work Copyright (c) 2012, Motorola Mobility, Inc. All Rights Reserved.

BSD License.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Motorola Mobility nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Additional contributions copyright their respective contributors (also BSD License):

* Fabrice Robinet
* Analytical Graphics, Inc.

The MIT License (MIT)

Copyright (c) 2014 KARASZI Istvn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Yasm is Copyright (c) 2001-2014 Peter Johnson and other Yasm developers.

Yasm developers and/or contributors include:
Peter Johnson
Michael Urman
Brian Gladman (Visual Studio build files, other fixes)
Stanislav Karchebny (options parser)
Mathieu Monnier (SSE4 instruction patches, NASM preprocessor additions)
Anonymous "NASM64" developer (NASM preprocessor fixes)
Stephen Polkowski (x86 instruction patches)
Henryk Richter (Mach-O object format)
Ben Skeggs (patches, bug reports)
Alexei Svitkine (GAS preprocessor)
Samuel Thibault (TASM parser and frontend)

Yasm licensing overview and summary
-----------------------------------

Note: This document does not provide legal advice nor is it the actual license of any part of Yasm. See the individual licenses for complete details. Consult a lawyer for legal advice.

The primary license of Yasm is the 2-clause BSD license. Please use this license if you plan on submitting code to the project.

Yasm has absolutely no warranty; not even for merchantibility or fitness for a particular purpose.

-------
Libyasm
-------
Libyasm is 2-clause or 3-clause BSD licensed, with the exception of bitvect, which is triple-licensed under the Artistic license, GPL, and LGPL. Libyasm is thus GPL and LGPL compatible. In addition, this also means that libyasm is free for binary-only distribution as long as the terms of the 3-clause BSD license and Artistic license (as it applies to bitvect) are fulfilled.
-------
Modules
-------
The modules are 2-clause or 3-clause BSD licensed.

-------
Frontends
-------
The frontends are 2-clause BSD licensed.

----------
License Texts
----------
The full text of all licenses are provided in separate files in the source distribution. Each source file may include the entire license (in the case of the BSD and Artistic licenses), or may reference the GPL or LGPL license file.

BSD.txt - 2-clause and 3-clause BSD licenses
Artistic.txt - Artistic license
GNU_GPL-2.0 - GNU General Public License
GNU_LGPL-2.0 - GNU Library General Public License
Copyright (C) 2005, 2006, 2007, 2008, 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE INC. AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE INC. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The "New" BSD License
************************

Copyright (c) 2004-2015, The Dojo Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Guillermo Rauch

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.
We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

   However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

   When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

   If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

   Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting
To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Copyright (C) 2016 by Marijn Haverbeke <marijnh@gmail.com> and others

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2016, Alliance for Open Media. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

language: node_js
node_js:
- '0.12'
- '4'
- '6'
- '7'
before_install:
- npm install -g npm@latest
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2008 Fair Oaks Labs, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

// Copyright (c) 2017 The Chromium OS Authors. All rights reserved.

// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

validate-npm-package-license

Give me a string and I'll tell you if it's a valid npm package license string.
var valid = require('validate-npm-package-license');
```

SPDX license identifiers are valid license strings:
```
```
var assert = require('assert');
var validSPDXExpression = {
  validForNewPackages: true,
  validForOldPackages: true,
  spdx: true
};

assert.deepEqual(valid('MIT'), validSPDXExpression);
assert.deepEqual(valid('BSD-2-Clause'), validSPDXExpression);
assert.deepEqual(valid('Apache-2.0'), validSPDXExpression);
assert.deepEqual(valid('ISC'), validSPDXExpression);
```

The function will return a warning and suggestion for nearly-correct license identifiers:
```
```
```
assert.deepEqual(
  valid('Apache 2.0'),
  {
    validForOldPackages: false,
    validForNewPackages: false,
    warnings: [
      'license should be ' +
      'a valid SPDX license expression (without "LicenseRef"), ' +
      '"UNLICENSED", or ' +
      '"SEE LICENSE IN <filename>"',
      'license is similar to the valid expression "Apache-2.0"'
    ]
  }
);
```
```

SPDX expressions are valid, too ...
```
```
```
```javascript

// Simple SPDX license expression for dual licensing
assert.deepEqual(
  valid('(GPL-3.0-only OR BSD-2-Clause)'),
  validSPDXExpression
);
```
... except if they contain `LicenseRef`:

```javascript
var warningAboutLicenseRef = {
  validForOldPackages: false,
  validForNewPackages: false,
  spdx: true,
  warnings: [
    'license should be ' +
    'a valid SPDX license expression (without "LicenseRef"), ' +
    '"UNLICENSED", or ' +
    '"SEE LICENSE IN <filename>"',
  ]};

assert.deepEqual(
  valid('LicenseRef-Made-Up'),
  warningAboutLicenseRef
);

assert.deepEqual(
  valid('(MIT OR LicenseRef-Made-Up)'),
  warningAboutLicenseRef
);

assert.deepEqual(
  valid('SEE LICENSE IN LICENSE.txt'),
  { validForNewPackages: true,
    validForOldPackages: true,
    inFile: 'LICENSE.txt' }
)

assert.deepEqual(
  valid('SEE LICENSE IN license.md'),
  { validForNewPackages: true,
    validForOldPackages: true,
    inFile: 'license.md' }
)
```
If there aren't any licensing terms, use `UNLICENSED`:

```javascript
```
var unlicensed = {
    validForNewPackages: true,
    validForOldPackages: true,
    unlicensed: true
};
assert.deepEqual(valid('UNLICENSED'), unlicensed);
assert.deepEqual(valid('UNLICENCED'), unlicensed);
```

APPLE PUBLIC SOURCE LICENSE
Version 2.0 - August 6, 2003

Please read this License carefully before downloading this software. By downloading or using this software, you are agreeing to be bound by the terms of this License. If you do not or cannot agree to the terms of this License, please do not download or use the software.

1. General; Definitions. This License applies to any program or other work which Apple Computer, Inc. ("Apple") makes publicly available and which contains a notice placed by Apple identifying such program or work as "Original Code" and stating that it is subject to the terms of this Apple Public Source License version 2.0 ("License"). As used in this License:

1.1 "Applicable Patent Rights" mean: (a) in the case where Apple is the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to Apple and (ii) that cover subject matter contained in the Original Code, but only to the extent necessary to use, reproduce and/or distribute the Original Code without infringement; and (b) in the case where You are the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to You and (ii) that cover subject matter in Your Modifications, taken alone or in combination with Original Code.

1.2 "Contributor" means any person or entity that creates or contributes to the creation of Modifications.

1.3 "Covered Code" means the Original Code, Modifications, the combination of Original Code and any Modifications, and/or any respective portions thereof.

1.4 "Externally Deploy" means: (a) to sublicense, distribute otherwise make Covered Code available, directly or indirectly, to anyone other than You; and/or (b) to use Covered Code, alone or as part of a Larger Work, in any way to provide a service, including but
not limited to delivery of content, through electronic communication with a client other than You.

1.5 "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.6 "Modifications" mean any addition to, deletion from, and/or change to, the substance and/or structure of the Original Code, any previous Modifications, the combination of Original Code and any previous Modifications, and/or any respective portions thereof. When code is released as a series of files, a Modification is: (a) any addition to or deletion from the contents of a file containing Covered Code; and/or (b) any new file or other representation of computer program statements that contains any part of Covered Code.

1.7 "Original Code" means (a) the Source Code of a program or other work as originally made available by Apple under this License, including the Source Code of any updates or upgrades to such programs or works made available by Apple under this License, and that has been expressly identified by Apple as such in the header file(s) of such work; and (b) the object code compiled from such Source Code and originally made available by Apple under this License.

1.8 "Source Code" means the human readable form of a program or other work that is suitable for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an executable (object code).

1.9 "You" or "Your" means an individual or a legal entity exercising rights under this License. For legal entities, "You" or "Your" includes any entity which controls, is controlled by, or is under common control with, You, where "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of fifty percent (50%) or more of the outstanding shares or beneficial ownership of such entity.

2. Permitted Uses; Conditions & Restrictions. Subject to the terms and conditions of this License, Apple hereby grants You, effective on the date You accept this License and download the Original Code, a world-wide, royalty-free, non-exclusive license, to the extent of Apple's Applicable Patent Rights and copyrights covering the Original Code, to do the following:

2.1 Unmodified Code. You may use, reproduce, display, perform, internally distribute within Your organization, and Externally Deploy verbatim, unmodified copies of the Original Code, for commercial or
non-commercial purposes, provided that in each instance:

(a) You must retain and reproduce in all copies of Original Code the copyright and other proprietary notices and disclaimers of Apple as they appear in the Original Code, and keep intact all notices in the Original Code that refer to this License; and

(b) You must include a copy of this License with every copy of Source Code of Covered Code and documentation You distribute or Externally Deploy, and You may not offer or impose any terms on such Source Code that alter or restrict this License or the recipients' rights hereunder, except as permitted under Section 6.

2.2 Modified Code. You may modify Covered Code and use, reproduce, display, perform, internally distribute within Your organization, and Externally Deploy Your Modifications and Covered Code, for commercial or non-commercial purposes, provided that in each instance You also meet all of these conditions:

(a) You must satisfy all the conditions of Section 2.1 with respect to the Source Code of the Covered Code;

(b) You must duplicate, to the extent it does not already exist, the notice in Exhibit A in each file of the Source Code of all Your Modifications, and cause the modified files to carry prominent notices stating that You changed the files and the date of any change; and

(c) If You Externally Deploy Your Modifications, You must make Source Code of all Your Externally Deployed Modifications either available to those to whom You have Externally Deployed Your Modifications, or publicly available. Source Code of Your Externally Deployed Modifications must be released under the terms set forth in this License, including the license grants set forth in Section 3 below, for as long as you Externally Deploy the Covered Code or twelve (12) months from the date of initial External Deployment, whichever is longer. You should preferably distribute the Source Code of Your Externally Deployed Modifications electronically (e.g. download from a web site).

2.3 Distribution of Executable Versions. In addition, if You Externally Deploy Covered Code (Original Code and/or Modifications) in object code, executable form only, You must include a prominent notice, in the code itself as well as in related documentation, stating that Source Code of the Covered Code is available under the terms of this License with information on how and where to obtain such Source Code.

2.4 Third Party Rights. You expressly acknowledge and agree that
although Apple and each Contributor grants the licenses to their respective portions of the Covered Code set forth herein, no assurances are provided by Apple or any Contributor that the Covered Code does not infringe the patent or other intellectual property rights of any other entity. Apple and each Contributor disclaim any liability to You for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, You hereby assume sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow You to distribute the Covered Code, it is Your responsibility to acquire that license before distributing the Covered Code.

3. Your Grants. In consideration of, and as a condition to, the licenses granted to You under this License, You hereby grant to any person or entity receiving or distributing Covered Code under this License a non-exclusive, royalty-free, perpetual, irrevocable license, under Your Applicable Patent Rights and other intellectual property rights (other than patent) owned or controlled by You, to use, reproduce, display, perform, modify, sublicense, distribute and Externally Deploy Your Modifications of the same scope and extent as Apple's licenses under Sections 2.1 and 2.2 above.

4. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In each such instance, You must make sure the requirements of this License are fulfilled for the Covered Code or any portion thereof.

5. Limitations on Patent License. Except as expressly stated in Section 2, no other patent rights, express or implied, are granted by Apple herein. Modifications and/or Larger Works may require additional patent licenses from Apple which Apple may grant in its sole discretion.

6. Additional Terms. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations and/or other rights consistent with the scope of the license granted herein ("Additional Terms") to one or more recipients of Covered Code. However, You may do so only on Your own behalf and as Your sole responsibility, and not on behalf of Apple or any Contributor. You must obtain the recipient's agreement that any such Additional Terms are offered by You alone, and You hereby agree to indemnify, defend and hold Apple and every Contributor harmless for any liability incurred by or claims asserted against Apple or such Contributor by reason of any such Additional Terms.
7. Versions of the License. Apple may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Once Original Code has been published under a particular version of this License, You may continue to use it under the terms of that version. You may also choose to use such Original Code under the terms of any subsequent version of this License published by Apple. No one other than Apple has the right to modify the terms applicable to Covered Code created under this License.

8. NO WARRANTY OR SUPPORT. The Covered Code may contain in whole or in part pre-release, untested, or not fully tested works. The Covered Code may contain errors that could cause failures or loss of data, and may be incomplete or contain inaccuracies. You expressly acknowledge and agree that use of the Covered Code, or any portion thereof, is at Your sole and entire risk. THE COVERED CODE IS PROVIDED "AS IS" AND WITHOUT WARRANTY, UPGRADES OR SUPPORT OF ANY KIND AND APPLE AND APPLE'S LICENSOR(S) (COLLECTIVELY REFERRED TO AS "APPLE" FOR THE PURPOSES OF SECTIONS 8 AND 9) AND ALL CONTRIBUTORS EXPRESSLY DISCLAIM ALL WARRANTIES AND/OR CONDITIONS, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY, OF SATISFACTORY QUALITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY, OF QUIET ENJOYMENT, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. APPLE AND EACH CONTRIBUTOR DOES NOT WARRANT AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE COVERED CODE, THAT THE FUNCTIONS CONTAINED IN THE COVERED CODE WILL MEET YOUR REQUIREMENTS, THAT THE OPERATION OF THE COVERED CODE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE COVERED CODE WILL BE CORRECTED. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY APPLE, AN APPLE AUTHORIZED REPRESENTATIVE OR ANY CONTRIBUTOR SHALL CREATE A WARRANTY. You acknowledge that the Covered Code is not intended for use in the operation of nuclear facilities, aircraft navigation, communication systems, or air traffic control machines in which case the failure of the Covered Code could lead to death, personal injury, or severe physical or environmental damage.

9. LIMITATION OF LIABILITY. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT SHALL APPLE OR ANY CONTRIBUTOR BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE OR YOUR USE OR INABILITY TO USE THE COVERED CODE, OR ANY PORTION THEREOF, WHETHER UNDER A THEORY OF CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCTS LIABILITY OR OTHERWISE, EVEN IF APPLE OR SUCH CONTRIBUTOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF ESSENTIAL PURPOSE OF ANY REMEDY. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OF LIABILITY OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS LIMITATION MAY NOT APPLY TO YOU. In no event shall Apple's total liability to You for all damages (other than as may be required by applicable law) under this
License exceed the amount of fifty dollars ($50.00).

10. Trademarks. This License does not grant any rights to use the trademarks or trade names "Apple", "Apple Computer", "Mac", "Mac OS", "QuickTime", "QuickTime Streaming Server" or any other trademarks, service marks, logos or trade names belonging to Apple (collectively "Apple Marks") or to any trademark, service mark, logo or trade name belonging to any Contributor. You agree not to use any Apple Marks in or as part of the name of products derived from the Original Code or to endorse or promote products derived from the Original Code other than as expressly permitted by and in strict compliance at all times with Apple's third party trademark usage guidelines which are posted at http://www.apple.com/legal/guidelinesfor3rdparties.html.

11. Ownership. Subject to the licenses granted under this License, each Contributor retains all rights, title and interest in and to any Modifications made by such Contributor. Apple retains all rights, title and interest in and to the Original Code and any Modifications made by or on behalf of Apple ("Apple Modifications"), and such Apple Modifications will not be automatically subject to this License. Apple may, at its sole discretion, choose to license such Apple Modifications under this License, or on different terms from those contained in this License or may choose not to license them at all.

12. Termination.

12.1 Termination. This License and the rights granted hereunder will terminate:

(a) automatically without notice from Apple if You fail to comply with any term(s) of this License and fail to cure such breach within 30 days of becoming aware of such breach;

(b) immediately in the event of the circumstances described in Section 13.5(b); or

(c) automatically without notice from Apple if You, at any time during the term of this License, commence an action for patent infringement against Apple; provided that Apple did not first commence an action for patent infringement against You in that instance.

12.2 Effect of Termination. Upon termination, You agree to immediately stop any further use, reproduction, modification, sublicensing and distribution of the Covered Code. All sublicenses to the Covered Code which have been properly granted prior to termination shall survive any termination of this License. Provisions which, by their nature, should remain in effect beyond the termination of this License shall survive, including but not limited to Sections 3, 5, 8, 9, 10, 11.
12.2 and 13. No party will be liable to any other for compensation, indemnity or damages of any sort solely as a result of terminating this License in accordance with its terms, and termination of this License will be without prejudice to any other right or remedy of any party.


13.1 Government End Users. The Covered Code is a "commercial item" as defined in FAR 2.101. Government software and technical data rights in the Covered Code include only those rights customarily provided to the public as defined in this License. This customary commercial license in technical data and software is provided in accordance with FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for Department of Defense purchases, DFAR 252.227-7015 (Technical Data -- Commercial Items) and 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation). Accordingly, all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

13.2 Relationship of Parties. This License will not be construed as creating an agency, partnership, joint venture or any other form of legal association between or among You, Apple or any Contributor, and You will not represent to the contrary, whether expressly, by implication, appearance or otherwise.

13.3 Independent Development. Nothing in this License will impair Apple's right to acquire, license, develop, have others develop for it, market and/or distribute technology or products that perform the same or similar functions as, or otherwise compete with, Modifications, Larger Works, technology or products that You may develop, produce, market or distribute.

13.4 Waiver; Construction. Failure by Apple or any Contributor to enforce any provision of this License will not be deemed a waiver of future enforcement of that or any other provision. Any law or regulation which provides that the language of a contract shall be construed against the drafter will not apply to this License.

13.5 Severability. (a) If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of the License will be enforced to the maximum extent permissible so as to effect the economic benefits and intent of the parties, and the remainder of this License will continue in full force and effect. (b) Notwithstanding the foregoing, if applicable law prohibits or restricts You from fully and/or specifically complying with Sections 2 and/or 3 or prevents the enforceability of either of those Sections, this License will
immediately terminate and You must immediately discontinue any use of
the Covered Code and destroy all copies of it that are in your
possession or control.

13.6 Dispute Resolution. Any litigation or other dispute resolution
between You and Apple relating to this License shall take place in the
Northern District of California, and You and Apple hereby consent to
the personal jurisdiction of, and venue in, the state and federal
courts within that District with respect to this License. The
application of the United Nations Convention on Contracts for the
International Sale of Goods is expressly excluded.

13.7 Entire Agreement; Governing Law. This License constitutes the
entire agreement between the parties with respect to the subject
matter hereof. This License shall be governed by the laws of the
United States and the State of California, except that body of
California law concerning conflicts of law.

Where You are located in the province of Quebec, Canada, the following
clause applies: The parties hereby confirm that they have requested
that this License and all related documents be drafted in English. Les
parties ont exige que le present contrat et tous les documents
connexes soient rediges en anglais.

EXHIBIT A.

"Portions Copyright (c) 1999-2003 Apple Computer, Inc. All Rights
Reserved.

This file contains Original Code and/or Modifications of Original Code
as defined in and that are subject to the Apple Public Source License
Version 2.0 (the 'License'). You may not use this file except in
compliance with the License. Please obtain a copy of the License at
http://www.opensource.apple.com/apsl/ and read it before using this
file.

The Original Code and all software distributed under the License are
distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT.
Please see the License for the specific language governing rights and
limitations under the License."

1.1418 parquet-format 2.3.0-incubating
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product depends on SLF4J and includes SLF4J in this binary artifact. SLF4J
is a simple logging facade for Java.

Copyright: 2004-2013 QOS.ch.
Home page: http://www.slf4j.org/
License: http://slf4j.org/license.html (MIT license)

The following is the SLF4J license (MIT):

Copyright (c) 2004-2013 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product depends on Apache Thrift and includes it in this binary artifact.
Copyright: 2006-2010 The Apache Software Foundation.
Home page: https://thrift.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

Apache Parquet Format (Incubating)
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1419 fast-deep-equal 3.1.3 3.1.3
1.1419.1 Available under license :
MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1420 @babel/plugin-transform-arrow-functions 7.16.0
1.1420.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1421 emoji-regex 8.0.0 8.0.0
1.1421.1 Available under license:
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1422 commons-text 1.9
1.1422.1 Available under license:

Apache Commons Text
Copyright 2014-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
Open Source Used In DNAC Platform Arica

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

**1.1423 ci-info 2.0.0**

**1.1423.1 Available under license :**

The MIT License (MIT)

Copyright (c) 2016-2018 Thomas Watson Steen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.1424 ignite-shmem 1.0.0**
1.1424.1 Available under license:

Apache License  
Version 2.0, January 2004  
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==============================================================================
Apache Ignite (incubating) Subcomponents:

The Apache Ignite project contains subcomponents with separate copyright
notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following
licenses.

==============================================================================
For the pcollections library (https://github.com/blackdrag/pcollections)

The MIT License (MIT)

Copyright (c) <year> <copyright holders>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
For SnapTree:

BSD LICENSE

Copyright (c) 2009 Stanford University, unless otherwise specified.
All rights reserved.

This software was developed by the Pervasive Parallelism Laboratory of Stanford University, California, USA.

Permission to use, copy, modify, and distribute this software in source or binary form for any purpose with or without fee is hereby granted, provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of Stanford University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1425 dicer 0.2.5 0.2.5

1.1425.1 Available under license:

Copyright Brian White. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1426 to-regex-range 5.0.1
1.1426.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1427 is-number 3.0.0 3.0.0
1.1427.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1428 path-exists 3.0.0 3.0.0

1.1428.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1429 number-is-nan 1.0.1 1.0.1

1.1429.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1430 httpcomponents-client 4.5.6

1.1430.1 Available under license:

Apache HttpClient
Copyright 1999-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1431 object-assign 4.1.0 4.1.0

1.1431.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1432 resolve 1.18.1 1.18.1

1.1432.1 Available under license:

MIT License

Copyright (c) 2012 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1433 avro-mapred 1.9.1

1.1433.1 Available under license:

Apache Avro
Copyright 2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Based upon the representations of upstream licensors, it is understood that
portions of the mapreduce API included in the Java implementation are licensed
from various contributors under one or more contributor license agreements to
Odiago, Inc. and were then contributed by Odiago to Apache Avro, which has now
made them available under the Apache 2.0 license. The original file header text
is:

| Licensed to Odiago, Inc. under one or more contributor license
| agreements. See the NOTICE file distributed with this work for
| additional information regarding copyright ownership. Odiago, Inc.
| licenses this file to you under the Apache License, Version 2.0
| (the "License"); you may not use this file except in compliance
| with the License. You may obtain a copy of the License at
| https://www.apache.org/licenses/LICENSE-2.0
|

| Unless required by applicable law or agreed to in writing, software
| distributed under the License is distributed on an "AS IS" BASIS,
| WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
| implied. See the License for the specific language governing
| permissions and limitations under the License.

The Odiago NOTICE at the time of the contribution:

| This product includes software developed by Odiago, Inc.
| (https://www.wibidata.com).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.1434 lodash.assign 4.2.0

1.1434.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/
Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own
licenses; we recommend you read them, as their terms may differ from the
terms above.

1.1435 mime 2.4.6 2.4.6
1.1435.1 Available under license:

The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1436 @babel/cli 7.16.0
1.1436.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1437 commons-logging 1.2

1.1437.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2006 The Apache Software Foundation.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9.jar/org/apache/commons/logging/LogFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2004,2006 The Apache Software Foundation.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/AvalonLogger.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2004 The Apache Software Foundation.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/WeakHashtable.java
No license file was found, but licenses were detected in source scan.

2004 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/package.html
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/package.html
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2004 The Apache Software Foundation.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/LogConfigurationException.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/Log4JLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/LogFactoryImpl.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/Jdk13LumberjackLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/SimpleLog.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/Jdk14Logger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/LogSource.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/NoOpLog.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/impl/LogKitLogger.java
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
  jar/org/apache/commons/logging/Log.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2005 The Apache Software Foundation.
*/
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/ServletContextCleaner.java

---


1.1438.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.1439 glob 6.0.4 6.0.4
**1.1439.1 Available under license:**

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

**1.1440 fill-range 4.0.0 4.0.0**

**1.1440.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.1441 babel-plugin-transform-es2015-sticky-regex 6.24.1**
1.1441.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHELL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1442

de.huxhorn.lilith.logback.appender.multiplex-classic 0.9.44

1.1442.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:
a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1443 jakarta xml bind api 2.3.2

1.1443.1 Available under license:

# Notices for Eclipse Project for JAXB

This content is produced and maintained by the Eclipse Project for JAXB project.

* Project home: https://projects.eclipse.org/projects/ee4jjaxb

## Trademarks

Eclipse Project for JAXB is a trademark of the Eclipse Foundation.

## Copyright
## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at http://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxb-api

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

Copyright (c) 2017, 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1444 assert-plus 1.0.0 1.0.0

1.1444.1 Available under license :

The MIT License (MIT)
Copyright (c) 2012 Mark Cavage

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1445 object.assign 4.1.1

1.1445.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1446 yargs-parser 18.1.3
1.1446.1 Available under license :
Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1447 prepend-http 1.0.4 1.0.4
1.1447.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1448 fastutil 6.5.7
1.1448.1 Available under license :
Found license 'GNU Lesser General Public License' in
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public License as published by the Free Software
Foundation; either version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, but WITH NOT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY * or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public
License for more details. You should have received a copy of the GNU Lesser General Public

1.1449 is-extglob 2.1.1 2.1.1
1.1449.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1450 uuid 3.3.2 3.3.2
1.1450.1 Available under license:
The MIT License (MIT)

Copyright (c) 2010-2016 Robert Kieffer and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1451 lm-sensors 3.4.0-2
1.1451.1 Available under license:
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

**GNU GENERAL PUBLIC LICENSE**
**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

  Gnomovision version 69, Copyright (C) year name of author
  Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
  This is free software, and you are welcome to redistribute it
  under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
  `Gnomovision' (which makes passes at compilers) written by James Hacker.

  <signature of Ty Coon>, 1 April 1989
  Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999
Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.
To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License").
Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY. TO THE EXTENT PERMITTED BY APPLICABLE LAW, EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Here is a list of the main contributors to lm-sensors version 3.

* Frodo Looijaard
  Original author of libsensors, sensors-detect, sensors and isadump.
* Merlin Hughes
  Original author of sensord.
* Bob Schlaermann
  Dynamic chip feature detection (a.k.a. generic chip support) in libsensors and sensors.
* Mark M. Hoffman
  Many improvements to the libsensors configuration file scanner.
* Jean Delivare
  New libsensors API, and migration of sensors and sensord thereto.
  Many optimizations in libsensors and sensors.
  Configuration file converter.
  Rewrite of sensors-detect.
  Support for multiple configuration files in libsensors.

1.1452 readable-stream 2.2.7 2.2.7
1.1452.1 Available under license:

Node.js is licensed for use as follows:

---

Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

===

This license applies to parts of Node.js originating from the
https://github.com/joyent/node repository:

===

Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

===

1.1453 jetty-webapp 9.4.34.v20201102

1.1453.1 Available under license:

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under
Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using
and distributing the Program and assumes all risks associated with its 
exercise of rights under this Agreement, including but not limited to the 
risks and costs of program errors, compliance with applicable laws, damage to 
or loss of data, programs or equipment, and unavailability or interruption of 
operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY 
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, 
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION 
LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN 
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) 
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE 
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY 
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under 
applicable law, it shall not affect the validity or enforceability of the 
remainder of the terms of this Agreement, and without further action by the 
parties hereto, such provision shall be reformed to the minimum extent 
necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Program itself 
(excluding combinations of the Program with other software or hardware) 
 infringes such Recipient's patent(s), then such Recipient's rights granted 
under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to 
comply with any of the material terms or conditions of this Agreement and does 
not cure such failure in a reasonable period of time after becoming aware of 
such noncompliance. If all Recipient's rights under this Agreement terminate, 
Recipient agrees to cease use and distribution of the Program as soon as 
reasonably practicable. However, Recipient's obligations under this Agreement 
and any licenses granted by Recipient relating to the Program shall continue 
and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in 
order to avoid inconsistency the Agreement is copyrighted and may only be 
modified in the following manner. The Agreement Steward reserves the right to 
publish new versions (including revisions) of this Agreement from time to 
time. No one other than the Agreement Steward has the right to modify this 
Agreement. The Eclipse Foundation is the initial Agreement Steward. The 
Eclipse Foundation may assign the responsibility to serve as the Agreement 
Steward to a suitable separate entity. Each new version of the Agreement will

Open Source Used In DNAC Platform Arica 6643
be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

====================================================================

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL/GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm
-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is
granted provided that the copyright notice appears in all copies.

1.1454 babel-helper-regex 6.26.0

1.1454.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1455 thrift 0.14.0

1.1455.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it.  You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price.  Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.
You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies
It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Tue Oct 24 12:28:44 CDT 2006

Copyright (c) <2006> <Martin J. Logan, Erlware>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software (OTP Base, fslib, G.A.S) and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This package was debianized by Thrift Developer's <dev@thrift.apache.org>.

This package and the Debian packaging is licensed under the Apache License, see `/usr/share/common-licenses/Apache-2.0`.

The following information was copied from Apache Thrift LICENSE file.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.
The following files contain some portions of code contributed under the Thrift Software License (see doc/old-thrift-license.txt), and relicensed under the Apache 2.0 License:

compiler/cpp/Makefile.am
compiler/cpp/src/generate/t_cpp_generator.cc
compiler/cpp/src/generate/t_netstd_generator.cc
compiler/cpp/src/generate/t_erl_generator.cc
compiler/cpp/src/generate/t_hs_generator.cc
compiler/cpp/src/generate/t_java_generator.cc
compiler/cpp/src/generate/t_ocaml_generator.cc
compiler/cpp/src/generate/t_perl_generator.cc
compiler/cpp/src/generate/t_php_generator.cc
compiler/cpp/src/generate/t_py_generator.cc
compiler/cpp/src/generate/t_rb_generator.cc
compiler/cpp/src/generate/t_st_generator.cc
compiler/cpp/src/generate/t_xsd_generator.cc
compiler/cpp/src/main.cc
compiler/cpp/src/parser/t_field.h
compiler/cpp/src/parser/t_program.h
compiler/cpp/src/platform.h
compiler/cpp/src/thriftl.ll
compiler/cpp/src/thrifty.yy
lib/netstd/src/Protocol/TBinaryProtocol.cs
lib/netstd/src/Protocol/TField.cs
lib/netstd/src/Protocol/TLi.st.cs
lib/netstd/src/Protocol/TMap.cs
lib/netstd/src/Protocol/TMessage.cs
lib/netstd/src/Protocol/TMessageT ype.cs
lib/netstd/src/Protocol/TProtocol.cs
lib/netstd/src/Protocol/TProtocolException.cs
lib/netstd/src/Protocol/TProtocolFactory.cs
lib/netstd/src/Protocol/TProtocolUtil.cs
lib/netstd/src/Protocol/TSet.cs
lib/netstd/src/Protocol/TStruct.cs
lib/netstd/src/Protocol/TTtype.cs
lib/netstd/src/Server/TServer.cs
lib/netstd/src/Server/TSimpleServer.cs
lib/netstd/src/Server/TTreadPoolServer.cs
lib/netstd/src/TApplicationException.cs
lib/netstd/src/Thrift.csproj
lib/netstd/src/Thrift.sln
lib/netstd/src/TProcessor.cs
lib/netstd/src/Transport/TServerSocket.cs
lib/netstd/src/Transport/TServerTransport.cs
lib/netstd/src/Transport/TSocket.cs
lib/netstd/src/Transport/TStreamTransport.cs
lib/netstd/src/Transport/TTransport.cs
lib/netstd/src/Transport/TTransportException.cs
lib/netstd/src/Transport/TTransportFactory.cs
lib/netstd/ThriftMSBuildTask/Properties/AssemblyInfo.cs
lib/netstd/ThriftMSBuildTask/ThriftBuild.cs
lib/netstd/ThriftMSBuildTask/ThriftMSBuildTask.csproj
lib/rb/lib/thrift.rb
lib/st/README
lib/st/thrift.st
test/OptionalRequiredTest.cpp
test/OptionalRequiredTest.thrift
test/ThriftTest.thrift

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

#   Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
#   Copying and distribution of this file, with or without
#   modification, are permitted in any medium without royalty provided
#   the copyright notice and this notice are preserved.

For the compiler/cpp/src/md5.[ch] components:

/*
   Copyright (C) 1999, 2000, 2002 Aladdin Enterprises.  All rights reserved.
   This software is provided 'as-is', without any express or implied
   warranty.  In no event will the authors be held liable for any damages
   arising from the use of this software.

   Permission is granted to anyone to use this software for any purpose,
   including commercial applications, and to alter it and redistribute it
   freely, subject to the following restrictions:

   1. The origin of this software must not be misrepresented; you must not
      claim that you wrote the original software. If you use this software
      in a product, an acknowledgment in the product documentation would be
      appreciated but is not required.
   2. Altered source versions must be plainly marked as such, and must not be
      misrepresented as being the original software.
   3. This notice may not be removed or altered from any source distribution.

   L. Peter Deutsch
   ghost@aladdin.com
*/
For the lib/rb/setup.rb: Copyright (c) 2000-2005 Minero Aoki,
lib/ocaml/OCamlMakefile and lib/ocaml/README-OCamlMakefile components:
   Copyright (C) 1999 - 2007 Markus Mottl

Licensed under the terms of the GNU Lesser General Public License 2.1
(see doc/lgpl-2.1.txt for the full terms of this license)
Apache Thrift
Copyright (C) 2006 - 2019, The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

# Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
# Copying and distribution of this file, with or without
# modification, are permitted in any medium without royalty provided
# the copyright notice and this notice are preserved.
For the lib/nodejs/lib/thrift/json_parse.js:

/*
json_parse.js
2015-05-02
Public Domain.
NO WARRANTY EXPRESSED OR IMPLIED. USE AT YOUR OWN RISK.
*/
(By Douglas Crockford <douglas@crockford.com>)

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

# Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
# Copying and distribution of this file, with or without
# modification, are permitted in any medium without royalty provided
# the copyright notice and this notice are preserved.

Open Source Used In DNAC Platform Arica 6677
For the lib/nodejs/lib/thrift/json_parse.js:

/*
json_parse.js
2015-05-02
Public Domain.
NO WARRANTY EXPRESSED OR IMPLIED. USE AT YOUR OWN RISK.
*/
(By Douglas Crockford <douglas@crockford.com>)

For lib/cpp/src/thrift/windows/SocketPair.cpp

/* socketpair.c
* Copyright 2007 by Nathan C. Myers <ncm@cantrip.org>; some rights reserved.
* This code is Free Software. It may be copied freely, in original or
* modified form, subject only to the restrictions that (1) the author is
* relieved from all responsibilities for any use for any purpose, and (2)
* this copyright notice must be retained, unchanged, in its entirety. If
* for any reason the author might be held responsible for any consequences
* of copying or use, license is withheld.
*/

For lib/py/compat/win32/stdint.h

// ISO C9x compliant stdint.h for Microsoft Visual Studio
// Based on ISO/IEC 9899-TC2 Committee draft (May 6, 2005) WG14/N1124
//
// Copyright (c) 2006-2008 Alexander Chemeris
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are met:
//
// 1. Redistributions of source code must retain the above copyright notice,
// this list of conditions and the following disclaimer.
//
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
//
// 3. The name of the author may be used to endorse or promote products
// derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED
// WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
// MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
// EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
// PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
// OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY.
// WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
// OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF
// ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
//
//////////////////////////////////////////////////////////////////////////////////

-------------------------------------------------------------------------
Codegen template in t_html_generator.h

* Bootstrap v2.0.3
* 
* Copyright 2012 Twitter, Inc
* Licensed under the Apache License v2.0
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Designed and built with all the love in the world @twitter by @mdo and @fat.

-------------------------------------------------------------------------
For t_cl_generator.cc

* Copyright (c) 2008- Patrick Collison <patrick@collison.ie>
* Copyright (c) 2006- Facebook

-------------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1456 junit 4.10
1.1456.1 Available under license : Common Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights
and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.
For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient’s rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient’s rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

1.1457 engine.io-client 1.8.5 1.8.5
1.1457.1 Available under license :
(The MIT License)

Copyright (c) 2014-2015 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1458 pako 1.0.11 1.0.11
1.1458.1 Available under license :
(The MIT License)

Copyright (C) 2014-2017 by Vitaly Puzrin and Andrei Tuputcyn

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1459 lodash 4.17.20 4.17.20

1.1459.1 Available under license:

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample
code is defined as all source code displayed within the prose of the
documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own
licenses; we recommend you read them, as their terms may differ from the
terms above.

1.1460 babel-helper-replace-supers 6.24.1

1.1460.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The "Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is
being made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
INTERNATIONAL.ORG/MEMENTO/CODEOFCONDUCT.HTM FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
Open Source Used In DNAC Platform Arica 6693

1.1461 junit-jupiter-api 5.7.0
1.1461.1 Available under license:

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

### 1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
  * **i)** changes to the Program, and
  * **ii)** additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.
Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

### 2. Grant of Rights

**a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

**b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

**c)** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
**d)** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

**e)** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

### 3. Requirements

**3.1** If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
  * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
  * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
  * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
  * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

**3.2** When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

**3.3** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

### 4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

### 5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

### 6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### 7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

#### Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.1462 aspectj-tools 1.8.13
1.1463

de.huxhorn.lilith.logback.appender.multiplex-core 0.9.44

1.1463.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:
a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this
license document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of
this License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would
otherwise be required to provide such information under section 6
of the GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the Combined
Work produced by recombining or relinking the Application with a
modified version of the Linked Version. (If you use option 4d0, the
Installation Information must accompany the Minimal Corresponding
Source and Corresponding Application Code. If you use option 4d1,
you must provide the Installation Information in the manner
specified by section 6 of the GNU GPL for conveying Corresponding
Source.)


You may place library facilities that are a work based on the
Library side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

/*
 * Lilith - a log event viewer.
 * Copyright (C) 2007-2011 Joern Huxhorn
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU Lesser General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public License
 * along with this program.  If not, see <http://www.gnu.org/licenses/>.
 */
1.1464 domhandler 2.4.2 2.4.2

1.1464.1 Available under license:

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1465 qs 6.7.0 6.7.0

1.1465.1 Available under license:

Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE
FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors

1.1466 guava 30.0-jre

1.1466.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 * * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of [@code Floats]. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/primitives/FloatsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/eventbus/Dispatcher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/eventbus/SubscriberRegistry.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/math/Quantiles.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/MoreObjects.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/eventbus/Subscriber.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/ListenerCallQueue.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/MathPreconditions.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ListeningScheduledExecutorService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/yang/CacheBuilderSpec.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/map/AbstractStreamingHasher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/AbstractCompositeHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/AbstractMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Present.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/BloomFilter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/primitives/UnsignedLongs.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/TreeRangeSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/AbstractHasher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingListeningExecutorService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/RegularImmutableMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/BloomFilterStrategies.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Optional.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Enums.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/BigIntegerMath.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/net/HostAndPort.java
Open Source Used In DNAC Platform Arica 6707

* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/ForwardingCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/primitives/Unsigned=Integer.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/CacheLoader.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/primitives/Unsigned=Integer.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/AbstractCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/hash/Funnels.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/collect/BoundType.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/cache/RemovalListeners.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/collect/AsyncFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-jre-sources-1.jar/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/base/Absent.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/FutureCallback.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/cache/MessageDigestHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collect/Queues.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/Uninterruptibles.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/hash/Hashing.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2005 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/reflect/Reflection.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/UncaughtExceptionHandlers.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/base/Strings.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/net/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/base/Equivalence.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/base/Strings.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/base/Ascii.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/annotations/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/annotations/Beta.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2009 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ImmutableSortedAsList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ImmutableSortedMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/javatime/Duration.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/ForwardingCondition.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/ForwardingLock.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of \{ @code Doubles\}. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/primitives/DoublesMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collect/MinMaxPriorityQueue.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collect/ForwardingImmutableCollection.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/AbstractSequentialIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ForwardingListMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/UnmodifiableListIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/RowSortedTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ForwardingSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/Synchronized.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ForwardingIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ClassToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/Multiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ImmutableList.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/AbstractIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/AbstractSortedSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/Synchronized.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ForwardingIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ClassToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/Multiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collection/ImmutableList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ComparatorOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/EnumBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/HashMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingConcurrentMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/UsingToStringOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/SortedSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ImmutableSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingCollection.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/EnumHashBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/LinkedHashMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/CompoundOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/AbstractMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingSortedMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/AbstractBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/AbstractListMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/SingletonImmutableSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/LinkedHashMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/NaturalOrdering.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ForwardingSortedSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/AbstractMapEntry.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/ToDoubleRounder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/Java8Compatibility.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is
distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied. See the License for the specific language governing permissions and limitations under
the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Streams.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ConsumingQueueIterator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ConsumingQueueIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AggregateFutureState.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/Platform.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/ReaderInputStream.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/LittleEndianByteArray.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2011 The Guava Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/package-info.java

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
 */
/*
 * This following method is a modified version of one found in
 * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
 * which contained the following notice:
 *
 * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
 * the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of [@code Shorts]. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/primitives/ShortsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of { @code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 The Guava Authors
 *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/Verify.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/io/CharSequenceReader.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/eventbus/SubscriberExceptionContext.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/DirectedNetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/NetworkBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/StandardMutableValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Comparators.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/MoreCollectors.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/LinkedHashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/StandardValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/GraphBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/ImmutableValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CollectCollectors.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/ValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/EndpointPair.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/UndirectedMultiNetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/UndirectedNetworkConnections.java

Open Source Used In DNAC Platform Arica  6721
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/StandardMutableGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/AbstractGraphBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/ElementOrder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collect/ImmutableSupplier.java
*/
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ExecutionSequencer.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/JdkBackedImmutableSet.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and

Open Source Used In DNAC Platform Arica  6723
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/RegularImmutableList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ComparisonChain.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/EmptyImmutableSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ComputationException.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ArrayTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/SingletonImmutableList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/RegularImmutableSortedSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/DiscreteDomain.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ForwardingTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableAsList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/SingletonImmutableTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableEnumSet.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2011 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

* Found in path(s):
  * /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/SortedMultiset.java
  * /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/SortedMultisets.java

No license file was found, but licenses were detected in source scan.

*/

Copyright (C) 2015 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
  * /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/ImmutableBiMapFauxverideShim.java
  * /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/graph/package-info.java
  * /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1.jar/com/google/common/collection/CollectSpliterators.java

No license file was found, but licenses were detected in source scan.

*/

Copyright (C) 2011 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

*/
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/

* This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
* and released to the public domain, as explained at
* http://creativecommons.org/licenses/publicdomain
* 
* As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
* java.util.HashMap class.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collect/TopKSelector.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/io/RecursiveDeleteOption.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/io/InsecureRecursiveDeleteException.java
*/
jar/com/google/common/graph/Network.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/Graphs.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/ImmutableGraph.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/ServiceManagerBridge.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/Internal.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/primitives/Platform.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/JdkBackedImmutableBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/IndexedImmutableSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/BaseImmutableMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/JdkBackedImmutableMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collect/GwtTransient.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 * *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/JdkPattern.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/PatternCompiler.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/CommonPattern.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/base/CommonMatcher.java
No license file was found, but licenses were detected in source scan.

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/AtomicDoubleArray.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/hash/Striped64.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/hash/LongAdder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collect/MultimapBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/collect/ImmutableMapEntry.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/io/MoreFiles.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/SparseImmutableTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Platform.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/Cut.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/DenseImmutableTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/RegularImmutableTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/MapMakerInternalMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/MapMaker.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/JdkFutureAdapters.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/graph/IncidentEdgeSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CompactHashing.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/Closer.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/RateLimiter.java
Open Source Used In DNAC Platform Arica 6734

* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/PairedStats.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/xml/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/html/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/WriteMode.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/math/StatsAccumulator.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/io/ByteStreams.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/base/Function.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/eventbus/Subscribe.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1
  jar/com/google/common/base/Objects.java
Open Source Used In DNAC Platform Arica 6736

* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/Closeables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/eventbus/AllowConcurrentEvents.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/Flushables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/Files.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Suppliers.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/Supplier.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/primitives/Primitives.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ListenableFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CollectPreconditions.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Table.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Range.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/UnmodifiableIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Platform.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableListMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableMapEntrySet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Collections2.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Serialization.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/PeekingIterator.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/HashBasedTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/StandardTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/SingletonImmutableBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableEntry.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/RegularImmutableBiMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableMapKeySet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/Tables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableCollection.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/TreeBasedTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/ImmutableSortedSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/StandardRowSortedTable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/ImmutableMapValues.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/RegularImmutableMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/EmptyImmutableListMultimap.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* /
**
* Holder for web specializations of methods of { @code Ints }. Intended to be empty for regular
* version.
* /

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/primitives/IntsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/ClosingFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/graph/Traverser.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
jar/com/google/common/util/concurrent/IgnoreJRERequirement.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/CompactLinkedHashMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/DescendingMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/FilteredMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/SortedSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/ForwardingSetMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/EvictingQueue.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/SortedMultisetBridge.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/AllEqualOrdering.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2006 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/DescendingImmutableSortedSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CompactHashSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CompactHashMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ForwardingBlockingDeque.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ForwardingImmutableList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/AbstractMultimap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableEnumMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ForwardingImmutableMap.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/CompactLinkedHashSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/TransformedListIterator.java

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/TimeoutFuture.java

Open Source Used In DNAC Platform Arica 6742
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractTimeLimiter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/util/concurrent/AggregateFuture.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Outer class that exists solely to let us write [@code Partially.GwtIncompatible] instead of plain
* [@code GwtIncompatible]. This is more accurate for [@link Futures#catching], which is available
* under GWT but with a slightly different signature.
*<p> We can't use [@code PartiallyGwtIncompatible] because then the GWT compiler wouldn't recognize
* it as a [@code GwtIncompatible] annotation. And for [@code Futures.catching], we need the GWT
* compiler to autostrip the normal server method in order to expose the special, inherited GWT
* version.
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/Partially.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* /

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collection/FluentIterable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/primitives/Floats.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/util/concurrent/SequentialExecutor.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/thirdparty/publicsuffix/TrieParser.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/primitives/Ints.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
/**
* Returns an array containing all of the elements in the specified collection. This method 
* returns the elements in the order they are returned by the collection's iterator. The returned 
* array is "safe" in that no references to it are maintained by the collection. The caller is 
* thus free to modify the returned array. 
* 
* <p>This method assumes that the collection size doesn't change while the method is running. 
* 
* <p>TODO(kevinb): support concurrently modified collections? 
* 
* @param c the collection for which to return an array of elements 
*/

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-
  jar/com/google/common/collection/ObjectArrays.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Not supported. <b>You are attempting to create a map that may contain a non-[@code Comparable]</b> 
* key.</b> Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy 
* version.
* 
* @throws UnsupportedOperationException always 
* @deprecated <b>Pass a key of type {@code Comparable} to use {link 
* ImmutableSortedMap#of(Comparable, Object)}.<b> 
*/
Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collect/ImmutableSortedMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/ImmutableSortedMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/SortedIterables.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/ForwardingSortedMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/RangeSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/GeneralRange.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/RegularImmutableSortedMultiset.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/AbstractRangeSet.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/SortedIterable.java
* /opt/cola/permits/1104665043_1604644117.38/0/guava-30-0-jre-sources-1-jar/com/google/common/collection/Count.java

1.1467 jetty-webapp 9.3.27.v20190418

1.1468 os-homedir 1.0.2
1.1468.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1469 duplexer2 0.1.4 0.1.4
1.1469.1 Available under license:
Copyright (c) 2013, Deoxxa Development

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Deoxxa Development nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY DEOXXA DEVELOPMENT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DEOXXA DEVELOPMENT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1470 object-keys 1.1.1 1.1.1
1.1470.1 Available under license:
The MIT License (MIT)

Copyright (C) 2013 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1471 immediate 3.0.6 3.0.6
1.1471.1 Available under license :
Copyright (c) 2012 Barnesandnoble.com, llc, Donavon West, Domenic Denicola, Brian Cavalier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1472 kite-hadoop-compatibility 1.1.0
1.1472.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 Cloudera Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446057_1631567977.53/0/kite-hadoop-compatibility-1-1-0-sources-
  jar/org/kitesdk/compat/Hadoop.java
* /opt/cola/permits/1202446057_1631567977.53/0/kite-hadoop-compatibility-1-1-0-sources-
  jar/org/kitesdk/compat/DynConstructors.java
* /opt/cola/permits/1202446057_1631567977.53/0/kite-hadoop-compatibility-1-1-0-sources-
  jar/org/kitesdk/compat/DynMethods.java

1.1473 drill-rdbms-metastore 1.19.0
1.1473.1 Available under license:

Drill : Metastore : RDBMS
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1474 libthai 0.1.24 2

1.1474.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999
Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
Open Source Used In DNAC Platform Arica 6758

programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based
on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that
you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact
all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the
Library.

You may charge a fee for the physical act of transferring a copy,
and you may at your option offer warranty protection in exchange for a
fee.

2. You may modify your copy or copies of the Library or any portion
of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices
stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no
charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a
table of data to be supplied by an application program that uses
the facility, other than as an argument passed when the facility
is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or
table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or
link a "work that uses the Library" with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least
three years, to give the same user the materials specified in
Subsection 6a, above, for a charge no more than the cost of
performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW,
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms
of the ordinary General Public License).

To apply these terms, attach the following notices to the library.
It is safest to attach them to the start of each source file to most
effectively convey the exclusion of warranty; and each file should
have at least the "copyright" line and a pointer to where the full
notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: libthai
Upstream-Contact: Theppitak Karoonboonyanan <thep@linux.thai.net>
Source: ftp://linux.thai.net/pub/thailinux/software/libthai/

Files: *
Copyright: 2001-2013 Theppitak Karoonboonyanan <thep@linux.thai.net>, Pattara Kiatisevi <ott@linux.thai.net>, Vuthichai Ampornaramveth <vuthi@nii.ac.jp>, Poonlap Veerathanabutr <poonlap@linux.thai.net>, Chanop Silpa-Anan <chanop@debian.org>
License: LGPL-2.1+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU Lesser General Public
License version 2.1 can be found in `/usr/share/common-licenses/LGPL-2.1'.

Files: debian/*
Copyright: 2003-2006 Pattara Kiatisevi <ott@linux.thai.net>
2006-2014 Theppitak Karoonboonyanan <thep@debian.org>
License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU General Public License
version 2 can be found in `/usr/share/common-licenses/GPL-2'.

1.1475 cglib 2.2.2
1.1475.1 Available under license :

========================================================================
== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0, ==
== in this case for the Apache Ant distribution. ==
========================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

This product includes also software developed by :
- the W3C consortium (http://www.w3c.org) ,
- the SAX project (http://www.saxproject.org)

Please read the different LICENSE files present in the root directory of
this distribution.
</DOCTYPE HTML PUBLIC "-/W3C//DTD HTML 4.0//EN">
<html>
Common Public License - v 1.0

The accompanying program is provided under the terms of this common public license ("agreement"). Any use, reproduction or distribution of the program constitutes recipient's acceptance of this agreement.

1. Definitions

"Contribution" means:

a) in the case of the initial contributor, the initial code and documentation distributed under this agreement, and
b) in the case of each subsequent contributor:

i) changes to the program, and

ii) additions to the program;

where such changes and/or additions to the program originate from and are distributed by that particular contributor. "Contribution" originates from a contributor if it was added to the program by such contributor itself or anyone acting on such contributor's behalf. Contributions do not include additions to the program which:

(i) are separate modules of software distributed in conjunction with the program under their own license agreement, and

(ii) are not derivative works of the program.

"Contributor" means any person or entity that distributes the program.

"Licensed Patents" mean patent claims licensable by a contributor which are necessarily infringed by the use or sale of its contribution alone or when combined with the program.

"Recipient" means anyone who receives the program under this agreement, including all contributors.
2. GRANT OF RIGHTS

Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:
<UL><FONT SIZE="2">a) it complies with the terms and conditions of this Agreement; and</FONT></UL>

<UL><FONT SIZE="2">b) its license agreement:</FONT></UL>

<UL><FONT SIZE="2">i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose; </FONT></UL>

<UL><FONT SIZE="2">ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;</FONT></UL>

<UL><FONT SIZE="2">iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and</FONT></UL>

<UL><FONT SIZE="2">iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.</FONT></UL>

<UL><FONT SIZE="2" COLOR="#0000FF">b) a copy of this Agreement must be included with each copy of the Program.<br>
</FONT></UL>

<UL><FONT SIZE="2" COLOR="#FF0000">Contributors may not remove or alter any copyright notices contained within the Program.</FONT></UL>

<UL><FONT SIZE="2">Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.</FONT></UL>

<UL><FONT SIZE="2">Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a

Open Source Used In DNAC Platform Arica 6770
commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent
applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that
Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In
addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a
lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware)
infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the
date such litigation is filed. <FONT SIZE="2"></FONT><p><FONT SIZE="2"></FONT>
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of
the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after
becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees
to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations
under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.
</FONT><p><FONT SIZE="2"></FONT>
Everyone is permitted to copy and distribute copies of this
Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the
following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Each Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Copyright (c) 2000, 2002, 2003 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
 *                                 Apache License
 *                           Version 2.0, January 2004
 *                        http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
 *
 * "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
 *
 * "Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
 *
 * "Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.
*
* 4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:
*
* (a) You must give any other recipients of the Work or
* Derivative Works a copy of this License; and
*
* (b) You must cause any modified files to carry prominent notices
* stating that You changed the files; and
*
* (c) You must retain, in the Source form of any Derivative Works
* that You distribute, all copyright, patent, trademark, and
* attribution notices from the Source form of the Work,
* excluding those notices that do not pertain to any part of
* the Derivative Works; and
*
* (d) If the Work includes a "NOTICE" text file as part of its
* distribution, then any Derivative Works that You distribute must
* include a readable copy of the attribution notices contained
* within such NOTICE file, excluding those notices that do not
* pertain to any part of the Derivative Works, in at least one
* of the following places: within a NOTICE text file distributed
* as part of the Derivative Works; within the Source form or
* documentation, if provided along with the Derivative Works; or,
* within a display generated by the Derivative Works, if and
* wherever such third-party notices normally appear. The contents
* of the NOTICE file are for informational purposes only and
* do not modify the License. You may add Your own attribution
* notices within Derivative Works that You distribute, alongside
* or as an addendum to the NOTICE text from the Work, provided
* that such additional attribution notices cannot be construed
* as modifying the License.
*
* You may add Your own copyright statement to Your modifications and
* may provide additional or different license terms and conditions
* for use, reproduction, or distribution of Your modifications, or
* for any such Derivative Works as a whole, provided Your use,
* reproduction, and distribution of the Work otherwise complies with
* the conditions stated in this License.
* 5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

* 6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

* 7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

* 8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

* 9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

* END OF TERMS AND CONDITIONS
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.1476 any-promise 1.3.0 1.3.0

1.1476.1 Available under license:

Copyright (C) 2014-2016 Kevin Beaty

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1477 array-flatten 1.1.1 1.1.1
1.1477.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1478 pify 4.0.1 4.0.1
1.1478.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1479 parsejson 0.0.3 0.0.3
1.1479.1 Available under license :
   The MIT License (MIT)
   
   Copyright (c) 2015 Gal Koren

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1480 code-point-at 1.1.0 1.1.0
1.1480.1 Available under license :
   The MIT License (MIT)

   Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1481 buffer-fill 1.0.0 1.0.0

1.1481.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1482 buffers 0.1.1 0.1.1

1.1482.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1483 micromatch 2.3.11

1.1483.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1484 to-array 0.1.4 0.1.4

1.1484.1 Available under license:

Copyright (c) 2012 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1485 is-descriptor 1.0.2 1.0.2

1.1485.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1486 batik-anim 1.13

1.1486.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/SVGOMFEOffsetElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/SVGOMAnimatedBoolean.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/AbstractAnimation.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/SVGOMSVGElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/XBLOMTemplateElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/timing/EventbaseTimingSpecifier.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/XBLOMImportElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/SVGOMDefinitionSrcElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/SVGOMDocument.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/org/apache/batik/anim/dom/
Open Source Used In DNAC Platform Arica 6791
jar/org/apache/batik/anim/values/AnimatableLengthListValue.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/timing/TimedElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/dom/SVGOMStyleElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/dom/SVGOMElement.java
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/dom/SVGOMAnimatedTransformList.java
No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/resources/Messages.properties
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/dom/resources/dtdids.properties
No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-
jar/org/apache/batik/anim/AnimationException.java
No license file was found, but licenses were detected in source scan.
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1146878560_1616835256.03/0/batik-anim-1-13-sources-1-jar/META-INF/maven/org.apache.xmlgraphics/batik-anim/pom.xml

1.1487 babel-helper-define-map 6.26.0

1.1487.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The "Software identified by reference to the Ecma Standard (("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA

CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrits noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ©COPYRIGHT HOLDER© BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1488 propagate 2.0.1

1.1488.1 Available under license:

The MIT License

Copyright (c) 2015-2019 Pedro Teixeira and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
1.1489 iconv-lite 0.4.24 0.4.24

1.1489.1 Available under license:

Copyright (c) 2011 Alexander Shtuchkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1490 @babel/plugin-syntax-async-generators 7.8.4

1.1490.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1491 kerb-server 1.0.0

1.1491.1 Available under license:

Kerby-kerb Server  
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at  
The Apache Software Foundation (http://www.apache.org/).

Apache License  
Version 2.0, January 2004  
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[[]]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1493 run-async 0.1.0

1.1493.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1494 serializator 1.0.2 1.0.2

1.1494.1 Available under license:

The MIT License (MIT)

Copyright (c) 2018 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1495 p-limit 1.3.0
1.1495.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1496 pstree-remy 1.1.8
1.1496.1 Available under license :
The MIT License (MIT)

Copyright 2019 Remy Sharp, https://remysharp.com <remy@remysharp.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1497 write-file-atomic 3.0.3
1.1497.1 Available under license :

Copyright (c) 2015, Rebecca Turner

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1498 jsreport-base 2.0.2 2.0.2
1.1498.1 Available under license :

MIT License

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1499 config 1.3.3

1.1499.1 Available under license:

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1500 events 2.1.0 2.1.0
1.1500.1 Available under license :
MIT

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the
following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1501 kerby-util 1.0.0

1.1501.1 Available under license:

Kerby Util
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1502 hadoop-yarn-common 3.2.2

1.1502.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveledb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:

* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4, which has the following notices:
The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
  * HOMEPAGE:
    * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
    * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
  * HOMEPAGE:
    * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
  * HOMEPAGE:
    * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
  * HOMEPAGE:
    * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
  * HOMEPAGE:
    * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
  * HOMEPAGE:
    * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
  * HOMEPAGE:
    * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:
This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
* https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:
* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

=================================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.
Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
Jetty Web Container
Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
 ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-------
Eclipse

The following artifacts are EPL.
  * org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
  * org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
  * org.eclipse.jetty.orbit:javax.mail.glassfish

-------
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

  * javax.servlet:javax.servlet-api
  * javax.annotation:javax.annotation-api
  * javax.transaction:javax.transaction-api
  * javax.websocket:javax.websocket-api
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL/GPLv2.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
PureJavaCrc32C from apache-hadoop-common http://hadoop.apache.org/ (Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1,
which has the following notices:
* == NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation
Licensed under the Apache License v2.0

--

Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:
--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* =============================================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  =============================================================================

  This product includes software developed by
  Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org
The binary distribution of this product bundles binaries of swagger-annotations (https://github.com/swagger-api/swagger-core), which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of metrics-core 3.2.4 which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

//*[++
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 * Abstract: The main routine
 * --*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/\{lz4.h,lz4.c,lz4hc.h,lz4hc.c\},

/*
   LZ4 - Fast LZ compression algorithm
   Header File
   Copyright (C) 2011-2014, Yann Collet.
   BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

   * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

---------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni (https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

============

The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following
license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

  * Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above
    copyright notice, this list of conditions and the following disclaimer
    in the documentation and/or other materials provided with the
distribution.
  * Neither the name of Google Inc. nor the names of its
    contributors may be used to endorse or promote products derived from
    this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
The binary distribution of this product bundles binaries of snappy
(http://code.google.com/p/snappy/), which is available under the following
license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
em-helpers v0.5.13
em-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.13.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodegs 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)
--------------------------------------------------------------------------------
Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this
file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the
License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
express or implied. See the License for the specific language governing permissions and
limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------
The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT
LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0

--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their
own licenses; we recommend you read them, as their terms may differ from the terms above.

For:
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js
--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js
--------------------------------------------------------------------------------

D3 is available under a 3-clause BSD license. For details, see:

hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:

HSQLDB Database 2.3.4

(HSQL License)

"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group.”

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

--------------------------------------------------------------------------------
(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0
1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

   (a) under intellectual property rights (other than
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section2.1(b) above, no patent license is granted: (1)for code that You delete from the Original Software, or (2)for infringements caused by: (i)the modification of the Original Software, or (ii)the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1)Modifications made by that Contributor (or portions thereof); and (2)the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section2.2(b) above, no patent license is granted: (1)for any code that Contributor has deleted from the Contributor Version; (2)for infringements caused by:
(i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License.
and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS,
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO
THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
gizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of
computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by
Contributor to use, reproduce, modify, display, perform, sublicense and
distribute the Modifications created by such Contributor (or portions thereof),
either on an unmodified basis, with other Modifications, as Covered Software
and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such
combination), to make, use, sell, offer for sale, have made, and/or otherwise
dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or
otherwise make available in Executable form must also be made available in
Source Code form and that Source Code form must be distributed only under the
terms of this License. You must include a copy of this License with every copy
of the Source Code form of the Covered Software You distribute or otherwise make
available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a
reasonable manner on or through a medium customarily used for software
exchange.
3.2. Modifications.
The Modifications that You create or to which
You contribute are governed by the terms of this License. You represent that You
believe Your Modifications are Your original creation(s) and/or You have
sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that
identifies You as the Contributor of the Modification. You may not remove or
alter any copyright, patent or trademark notices contained within the Covered
Software, or any notices of licensing or any descriptive text giving attribution
to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source
Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.
When You are an
Initial Developer and You want to create a new license for Your Original
Software, You may create and use a modified version of this License if You: (a)
rename the license and remove any references to the name of the license steward
(except to note that the license differs from this License); and (b) otherwise
make it clear that the license contains terms which differ from this
License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS
LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE
IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR
NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED
SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY
RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE
COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF
WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED
SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will
terminate automatically if You fail to comply with terms herein and fail to cure
such breach within 30 days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination of this License shall
survive.

6.2. If You assert a patent infringement claim (excluding declaratory
judgment actions) against Initial Developer or a Contributor (the Initial
Developer or Contributor against whom You assert such claim is referred to as
Participant) alleging that the Participant Software (meaning the
Contributor Version where the Participant is a Contributor or the Original
Software where the Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or indirectly to
You by such Participant, the Initial Developer (if the Initial Developer is not
the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this
License shall, upon 60 days notice from Participant terminate prospectively and
automatically at the expiration of such 60 day notice period, unless if within
such 60 day period You withdraw Your claim with respect to the Participant
Software against such Participant either unilaterally or pursuant to a written
agreement with Participant.

6.3. If You assert a patent infringement claim
against Participant alleging that the Participant Software directly or
indirectly infringes any patent where such claim is resolved (such as by license
or settlement) prior to the initiation of patent infringement litigation, then
the reasonable value of the licenses granted by such Participant under Sections
2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.
6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software.
Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:

Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in src/google/protobuf/stubs/atomicops_internals_generic_gcc.h. This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in src/google/protobuf/stubs/atomicops_internals_power.h. This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04
--------------------------------------------------------------------------------

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1
--------------------------------------------------------------------------------
(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are not
derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free copyright license to
   reproduce, prepare derivative works of, publicly display, publicly perform,
   distribute and sublicense the Contribution of such Contributor, if any, and such
derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
   Patents to make, use, sell, offer to sell, import and otherwise transfer the
   Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other combinations
which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to
   its Contributions set forth herein, no assurances are provided by any
   Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties of title and non-infringement, and implied warranties of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if
a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL
If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the
following license:
"Java Concurrency in Practice" book annotations 1.0
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS
PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR
OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE
BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED
HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or
encyclopedia, in which the Work in its entirety in unmodified form, along with a
number of other contributions, constituting separate and independent works in
themselves, are assembled into a collective whole. A work that constitutes a
Collective Work will not be considered a Derivative Work (as defined below) for
the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other
pre-existing works, such as a translation, musical arrangement, dramatization,
fictionalization, motion picture version, sound recording, art reproduction,
abridgment, condensation, or any other form in which the Work may be recast,
transformed, or adapted, except that a work that constitutes a Collective Work
will not be considered a Derivative Work for the purpose of this License. For
the avoidance of doubt, where the Work is a musical composition or sound
recording, the synchronization of the Work in timed-relation with a moving image
(“synching”) will be considered a Derivative Work for the purpose of this
License.

"Licensor" means the individual or entity that offers the Work under the terms
of this License.
"Original Author" means the individual or entity who created the Work.
"Work" means the copyrightable work of authorship offered under the terms of this License.
"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
to create and reproduce Derivative Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.
For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.
Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).
Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).
The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats.
All rights not expressly granted by Licensor are hereby reserved.
4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY,
OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
The binary distribution of this product bundles these dependencies under the following license:
jamon-runtime 2.4.1

--------------------------------------------------------------------------------

(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the
time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have
made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third partys modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions
Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known
3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been
validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.
10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If you choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

/*

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

--------------------------------------------------------------------------------

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a
Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image (“synching”) will be considered an Adaptation for the purpose of this License.

b. “Collection” means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. “Distribute” means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. “Licensor” means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. “Original Author” means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. “Work” means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise
considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish." or a modification could indicate "The original work has been modified.";

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or
compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. You may not sub license the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and
provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY
KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing.
and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.

--------------------------------------------------------------------------------

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1503 lazy 1.0.11 1.0.11

1.1503.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1504 regexpu-core 4.8.0

1.1504.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1505 p-is-promise 2.1.0 2.1.0

1.1505.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache HttpCore
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1507 babel-plugin-check-es2015-constants
6.22.0 6.22.0

1.1507.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1511 guice 5.0.1
1.1511.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1512 is-resolvable 1.1.0

1.1512.1 Available under license :

ISC License (ISC)
Copyright 2018 Shinnosuke Watanabe

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1513 babel-register 7.16.0

1.1513.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1514 drill-metastore-api 1.19.0

1.1514.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided...
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Drill : Metastore : API
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1515 denque 1.5.0 1.5.0
1.1515.1 Available under license :
Copyright (c) 2018 Mike Diarmid (Salakar) <mike.diarmid@gmail.com>

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this library except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1516 utf-8-validate 1.2.1 1.2.1

1.1516.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1517 supports-color 6.0.0

1.1517.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1518 fast-deep-equal 2.0.1 2.0.1

1.1518.1 Available under license:
MIT License

Copyright (c) 2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1519 event-emitter 0.3.5

1.1519.1 Available under license:
Copyright (C) 2012-2015 Mariusz Nowak (www.medikoo.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1520 jctools-core 3.1.0

1.1520.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/spec/Ordering.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/SpscUnboundedAtomicArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/SpmcAtomicArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/MessagePassingQueueUtil.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/MpmcAtomicArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/util/UnsafeJvmInfo.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/LinkedQueueAtomicNode.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/SpscLinkedQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources-
jar/org/jctools/queues/atomic/MpscGrowableAtomicArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/maps/NonBlockingIdentityHashMap.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/BaseLinkedQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/util/InternalAPI.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/maps/NonBlockingHashSet.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/LinkedQueueNode.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/MpscLinkedQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/maps/NonBlockingSetInt.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/atomic/BaseLinkedAtomicQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/atomic/SequencedAtomicReferenceArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/MpscCompoundQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/util/UnsafeRefArrayAccess.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/MpscChunkedArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/SpscArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/MpscLinkedQueue.java
* /opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/jctools-core-3-1-0-sources-
arica.jar/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java
jar/org/jctools/queues/QueueFactory.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/MpscArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/MpscUnboundedArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/atomic/SpscGrowableAtomicArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/IndexedQueueSizeUtil.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/atomic/MpscLinkedAtomicQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/SpscGrowableArrayQueue.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/util/RangeUtil.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/package-info.java
*/opt/ws_local/PERMITS_SQL/1093815924_1601014602.33/0/jctools-core-3-1-0-sources.jar/org/jctools/queues/BaseSpscLinkedArrayQueue.java
No license file was found, but licenses were detected in source scan.

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.jctools</groupId>
  <artifactId>jctools-core</artifactId>
  <version>3.1.0</version>
  <name>Java Concurrency Tools Core Library</name>
  <description>Java Concurrency Tools Core Library</description>
  <packaging>bundle</packaging>
  <dependencies>
    <dependency>
      <groupId>org.hamcrest</groupId>
      <artifactId>hamcrest-all</artifactId>
      <version>${hamcrest.version}</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
<dependency>
<groupId>com.google.guava</groupId>
<artifactId>guava-testlib</artifactId>
<version>${guava-testlib.version}</version>
<scope>test</scope>
</dependency>
</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-surefire-plugin</artifactId>
<version>3.0.0-M3</version>
<configuration>
<includes>
<include>*</include>
</includes>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.felix</groupId>
<artifactId>maven-bundle-plugin</artifactId>
<version>4.2.1</version>
<extensions>true</extensions>
<configuration>
<instructions>
<Import-Package>sun.misc;resolution:=optional</Import-Package>
</instructions>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
<version>3.2.0</version>
<executions>
<execution>
<id>attach-sources</id>
<phase>verify</phase>
<goals>
<goal>jar-no-fork</goal>
</goals>
</execution>
</executions>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
<version>3.1.1</version>
<configuration>
<includes>
<include>*</include>
</includes>
</configuration>
</plugin>
</plugins>
</build>
1.1521 babel-plugin-syntax-exponentiation-operator 6.13.0

1.1521.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1522 asm 7.3.1

1.1522.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/FieldVisitor.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Edge.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Handler.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Constants.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/RecordComponentWriter.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/RecordComponentVisitor.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/AnnotationVisitor.java
jar/org/objectweb/asm/ConstantDynamic.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/ModuleWriter.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/CurrentFrame.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/ModuleVisitor.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Opcodes.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Attribute.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/MethodWriter.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/Handle.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/MethodTooLargeException.java
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/signature/package.html

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135907460_1613628459.33/0/asm-7-3-1-sources-2-jar/org/objectweb/asm/signature/package.html

---

Open Source Used In DNAC Platform Arica  6923
1.1523 utils-merge 1.0.0 1.0.0
1.1523.1 Available under license:
(The MIT License)

Copyright (c) 2013 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1524 bytes 3.1.0 3.1.0
1.1524.1 Available under license:
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Jed Watson <jed.watson@me.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1525 mime-types 2.1.27 2.1.27
1.1525.1 Available under license :
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1526 paranamer 2.5.6

1.1527 opentracing-api 0.31.0
1.1527.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.1528 snappy 1.0.5

1.1528.1 Available under license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===

Some of the benchmark data in testdata/ is licensed differently:
- fireworks.jpeg is Copyright 2013 Steinar H. Gunderson, and is licensed under the Creative Commons Attribution 3.0 license (CC-BY-3.0). See https://creativecommons.org/licenses/by/3.0/ for more information.

- kppkn.gtb is taken from the Gaviota chess tablebase set, and is licensed under the MIT License. See https://sites.google.com/site/gaviotachessengine/Home/endgame-tablebases-1 for more information.

- paper-100k.pdf is an excerpt (bytes 92160 to 194560) from the paper Combinatorial Modeling of Chromatin Features Quantitatively Predicts DNA Replication Timing in _Drosophila_ by Federico Comoglio and Renato Paro, which is licensed under the CC-BY license. See http://www.ploscompbiol.org/static/license for more information.

- alice29.txt, asyoulik.txt, plrabn12.txt and lcet10.txt are from Project Gutenberg. The first three have expired copyrights and are in the public domain; the latter does not have expired copyright, but is still in the public domain according to the license information (http://www.gutenberg.org/ebooks/53).

1.1529 brace-expansion 1.1.11 1.1.11

1.1529.1 Available under license :

MIT License

Copyright (c) 2013 Julian Gruber <julian@juliangruber.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1530 lodash.isundefined 3.0.1 3.0.1

1.1530.1 Available under license:

Copyright 2012-2015 The Dojo Foundation <http://dojofoundation.org/>
Based on Underscore.js, copyright 2009-2015 Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1531 @babel/plugin-syntax-logical-assignment-operators 7.10.4

1.1531.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1532 @babel/traverse 7.11.5
1.1532.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1533 jasper-compiler 5.5.12
1.1533.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1534 safe-buffer 5.2.1 5.2.1

1.1534.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1535 after 0.8.2 0.8.2
1.1535.1 Available under license:
Copyright (c) 2011 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1536 antlr 3.5.2
1.1536.1 Available under license:
BSD licence
http://antlr.org/license.html

1.1537 avatica-core 1.15.0
1.1537.1 Available under license:
Apache Calcite Avatica
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1538 protobuf-java-util 3.13.0

1.1538.1 Available under license:

No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
//   notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
//   copyright notice, this list of conditions and the following disclaimer
//   in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
//   this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/Timestamps.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/Durations.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/Values.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/FieldMaskUtil.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/FieldMaskTree.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/Structs.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/TimeUtil.java
* /opt/cola/permits/1258286292_1643052491.54/0/protobuf-java-util-3-13-0-sources-jar/com/google/protobuf/util/JsonFormat.java

1.1539 utile 0.3.0 0.3.0

1.1539.1 Available under license:

Copyright (c) 2011, Charlie Robbins & the Contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1540  fs.realpath  1.0.0  1.0.0
1.1540.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----

This library bundles a version of the `fs.realpath` and `fs.realpathSync` methods from Node.js v0.10 under the terms of the Node.js MIT license.

Node's license follows, also included at the header of `old.js` which contains the licensed code:

Copyright Joyent, Inc. and other Node contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1541 amcharts3-responsive 1.0.5 1.0.5
1.1541.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{"}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1542 socket.io-parser 2.3.1 2.3.1

1.1542.1 Available under license:
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1543 slf4j 1.7.2

1.1543.1 Available under license:
Copyright (c) 2004-2008 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1544 metrics-core 2.2.0

1.1544.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/main/java/util/concurrent/ThreadLocalRandom.java?view=markup
 */

Found in path(s):
* /opt/cola/permits/1145440743_1616178883.36/0/metrics-core-2-2-0-sources-5-jar/com/yammer/metrics/stats/ThreadLocalRandom.java
1.1545 jersey-media-json-jackson 2.25.1

1.1545.1 Available under license:

CDDL+GPL License
http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1546 chalk 0.4.0 0.4.0

1.1546.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1547 on-headers 1.0.1 1.0.1

1.1547.1 Available under license:

(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERChANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1548 lodash.isplainobject 4.0.6 4.0.6

1.1548.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1549 jakarta-inject 2.6.1
1.1549.1 Available under license:
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.
"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the
licenses to its Contributions set forth herein, no assurances are
provided by any Contributor that the Program does not infringe the
patent or other intellectual property rights of any other entity.
Each Contributor disclaims any liability to Recipient for claims
brought by any other entity based on infringement of intellectual
property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby
assumes sole responsibility to secure any other intellectual
property rights needed, if any. For example, if a third party
patent license is required to allow Recipient to Distribute the
Program, it is Recipient's responsibility to acquire that license
before distributing the Program.

d) Each Contributor represents that to its knowledge it has
sufficient copyright rights in its Contribution, if any, to grant
the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no
Contributor makes additional grants to any Recipient (other than
those set forth in this Agreement) as a result of such Recipient's
receipt of the Program under the terms of a Secondary License
(if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in
accordance with section 3.2, and the Contributor must accompany
the Program with a statement that the Source Code for the Program
is available under this Agreement, and informs Recipients how to
obtain it in a reasonable manner on or through a medium customarily
used for software exchange; and

b) the Contributor may Distribute the Program under a license
different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all
      warranties and conditions, express and implied, including
      warranties or conditions of title and non-infringement, and
      implied warranties or conditions of merchantability and fitness
      for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all
      liability for damages, including direct, indirect, special,
      incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights
      in the Source Code under section 3.2; and
iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial
product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's
rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to
look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided
that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made
available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.1550 is-accessor-descriptor 0.1.6

#### 1.1550.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1551 shelljs 0.8.4 0.8.4

1.1551.1 Available under license:
Copyright (c) 2012, Artur Adib <arturadib@gmail.com>
All rights reserved.

You may use this project under the terms of the New BSD license as follows:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Artur Adib nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ARTUR ADIB BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1552 query-string 7.0.0 7.0.0
1.1552.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (http://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1553 fontconfig 2.11.94 0ubuntu1.1
1.1553.1 Available under license:

This package was debianized by Colin Walters <walters@debian.org> on Sun, 13 Oct 2002 15:01:50 -0400

It was downloaded from http://www.fontconfig.org/

Upstream Author: Keith Packard

Copyright:

Copyright 2001,2003 Keith Packard

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Keith Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Keith Packard makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

KEITH PACKARD DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL KEITH PACKARD BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

fontconfig/COPYING

Copyright 2005 Patrick Lam
Copyright 2009 Roozbeh Pournader
Copyright 2008,2009 Red Hat, Inc.
Copyright 2008 Danilo egan
Copyright 2012 Google, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of the author(s) not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The authors make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THE AUTHOR(S) DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1554 chokidar 3.3.0

1.1554.1 Available under license :

The MIT License (MIT)

Copyright (c) 2012-2019 Paul Miller (https://paulmillr.com), Elan Shanker

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1555 nodemon 2.0.15

1.1555.1 Available under license :
(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2012 TJ Holowaychuk
Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Open Source Used In DNAC Platform Arica  6971
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2009-2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

// MIT License

Copyright (C) Roman Shtylman <shtylman@gmail.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2013-2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2013 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in the software.

Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright holders and contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors
The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2010 - present, Remy Sharp, https://remysharp.com <remy@remysharp.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright 2014 Alex Gorbatchev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Open Source Used In DNAC Platform Arica 6978

(The MIT License)

Copyright (c) 2012 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2012 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1556 lodash-set 4.3.2 4.3.2

1.1556.1 Available under license:
Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1557 slice-ansi 0.0.4
1.1557.1 Available under license :
(The MIT License)

Copyright (c) 2015 DC <threedeecee@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1558 commons-configuration 1.6
1.1558.1 Available under license :
Apache Commons Configuration
Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1559 pdfjs-dist 2.0.489 2.0.489

1.1559.1 Available under license:

%%Copyright: --------------------------------------------------------------------------------------------------
%%Copyright: Copyright 1990-2009 Adobe Systems Incorporated.
%%Copyright: All rights reserved.
%%Copyright:
%%Copyright: Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
%%Copyright: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
%%Copyright:
%%Copyright: Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
%%Copyright: Neither the name of Adobe Systems Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
1.1560 babel-plugin-check-es2015-constants

6.22.0

1.1560.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXpress OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY. WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1561 defer-to-connect 1.1.3

1.1561.1 Available under license :

MIT License

Copyright (c) 2018 Szymon Marczak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1562 jsreport-static-pdf 0.4.0 0.4.0

1.1562.1 Available under license :

MIT License

Copyright (c) 2019 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1563 buffer-alloc 1.2.0 1.2.0

1.1563.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1564 nanoid 3.1.12 3.1.12
1.1564.1 Available under license:

The MIT License (MIT)

Copyright 2017 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1565 normalize-url 4.5.0

1.1565.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1566 graphics2d 0.30

1.1566.1 Available under license:

Copyright (c) 2011-12, vernon adams (vern@newtypography.co.uk), with Reserved Font Names 'Antonio'

This Font Software is licensed under the SIL Open Font License, Version 1.1.
This license is copied below, and is also available with a FAQ at: http://scripts.sil.org/OFL

-----------------------------------------------------------
SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007
-----------------------------------------------------------

PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software,
provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1567 wordwrap 1.0.0
1.1567.1 Available under license :

   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2018 Colin Arenz

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1568 jsreport-assets 1.7.0 1.7.0
1.1568.1 Available under license :
MIT License

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1569 cookie 0.1.5 0.1.5
1.1569.1 Available under license :
(The MIT License)

Copyright (c) 2012-2014 Roman Shtylman <shtylman@gmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1570 ecdsa-sig-formatter 1.0.10 1.0.10

1.1570.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 D2L Corporation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1571 jackson-jaxrs 2.12.1

1.1571.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.salaranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

---

### 1.1572 ms 2.0.0 2.0.0

1.1572.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1573 p-defer 1.0.0 1.0.0

1.1573.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.1574 unc-path-regex 0.1.2 0.1.2

1.1574.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.1575 dom 1.0
1.1575.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2001 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University). All
* Rights Reserved. This program is distributed under the W3C's Software
* Intellectual Property License. This program is distributed in the
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
* PURPOSE.
* See W3C License http://www.w3.org/Consortium/Legal/ for more details.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ElementEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASNotationDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASEntityDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASNamedObjectMap.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/CharacterDataEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASObjectList.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/DocumentAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASEntityDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASAttributeDeclaration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/DocumentEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/DocumentEditAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASObject.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/DOMImplementationAS.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASDataType.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASContentModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom3/as/ASObjectList.java
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/DOMASBuilder.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/DOMASException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom3/as/NodeEditAS.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2000 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University). All
* Rights Reserved. This program is distributed under the W3C's Software
* Intellectual Property License. This program is distributed in the
* hope that it will be useful, but WITHOUT ANY WARRANTY; without even
* the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
* PURPOSE. See W3C License http://www.w3.org/Consortium/Legal/ for more
* details.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/w3c/dom/html/HTMLDOMImplementation.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * Checks if this content model has had its min/maxOccurs values reduced for
 * purposes of speeding up UPA. If so, this content model should not be used
 * for any purpose other than checking unique particle attribution
 *
 * @return a boolean that says whether this content has been compacted for UPA
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/models/XSCMValidator.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * DOM Level 3 LS CR - Experimental.
 * Create a new <code>LSParser</code>. The newly constructed parser may
 * then be configured by means of its <code>DOMConfiguration</code>
 * object, and used to parse documents by means of its <code>parse</code>
* method.
* @param mode The <code>mode</code> argument is either
* <code>MODE_SYNCHRONOUS</code> or <code>MODEASYNCHRONOUS</code>, if
* <code>mode</code> is <code>MODE_SYNCHRONOUS</code> then the
* <code>LSParser</code> that is created will operate in synchronous
* mode, if it's <code>MODEASYNCHRONOUS</code> then the
* <code>LSParser</code> that is created will operate in asynchronous
* mode.
* @param schemaType An absolute URI representing the type of the schema
* language used during the load of a <code>Document</code> using the
* newly created <code>LSParser</code>. Note that no lexical checking
* is done on the absolute URI. In order to create a
* <code>LSParser</code> for any kind of schema types (i.e. the
* LSParser will be free to use any schema found), use the value
* <code>null</code>.
* <p><b>Note:</b> For W3C XML Schema [a href='http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/']XML Schema Part 1</a>]
* , applications must use the value
* <code>http://www.w3.org/2001/XMLSchema"</code>. For XML DTD [a href='http://www.w3.org/TR/2000/REC-xml-20001006']XML 1.0</a>,
* applications must use the value
* <code>http://www.w3.org/TR/REC-xml"</code>. Other Schema languages
* are outside the scope of the W3C and therefore should recommend an
* absolute URI in order to use this method.
* @return The newly created <code>LSParser</code> object. This
* <code>LSParser</code> is either synchronous or asynchronous
* depending on the value of the <code>mode</code> argument.
* <p><b>Note:</b> By default, the newly created <code>LSParser</code>
* does not contain a <code>DOMErrorHandler</code>, i.e. the value of
* the "<a href='http://www.w3.org/TR/2003/WD-DOM-Level-3-Core-20030609/core.html#parameter-error-handler'>error-handler</a>" configuration parameter is <code>null</code>. However, implementations
* may provide a default error handler at creation time. In that case,
* the initial value of the <code>error-handler</code> configuration
* parameter on the new created <code>LSParser</code> contains a
* reference to the default error handler.
* @exception DOMException
* NOT_SUPPORTED_ERR: Raised if the requested mode or schema type is
* not supported.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/CoreDOMImplementationImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom/DOMImplementationListImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/**
 * used to check the 3 constraints against each complex type
 * (should be each model group):
 * Unique Particle Attribution, Particle Derivation (Restriction),
 * Element Declarations Consistent.
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSConstraints.java
No license file was found, but licenses were detected in source scan.
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
*      http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDocumentInfo.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDElementTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/parser/XMLErrorHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLFrameElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDAbstractIDConstraintTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLImgElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/parser/XMLInputSource.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/WMLOptgroupElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/parser/DeferredCDATASectionImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLWmlElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/xs/AnySimpleDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/defaults/StringList.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/WMLTimerElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/regex/CaseInsensitiveMap.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/opti/SchemaDOMImplementation.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/opti/DefaultXMLDocumentHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/WMLAnchorElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xml/serialize/HTMLSerializer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLHtmlElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xinclude/MultipleScopeNamespaceSupport.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/opti/SchemaParsingConfig.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dtd/XML11NSDTDValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xpointer/XPointerMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xs/XSException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLMetaElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLQuoteElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/util/EncodingMap.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xinc/TextReader.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/DOMInputImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/ObjectFactory.java
Open Source Used In DNAC Platform Arica  7022

* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/XSSimpleTypeDelegate.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/opti/SchemaDOMParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xpath/regex/RegularExpression.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/opti/ElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/events/UEventImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xml/serialize/Method.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/stax/events/EntityReferenceImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/html/dom/HTMLBodyElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dtd/XMLNSDTDValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xx/datatypes/ByteList.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dt/dt/XMLNSValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xpointer/ElementSchemePointer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/msg/XMLMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xx/models/XSCMBinOp.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/DeferredAttrImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/SAXParserFactoryImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xpointer/ElementSchemePointer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/msg/XMLMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/opti/DefaultElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/SofReferenceGrammarPool.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/AbstractXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/AttrNSImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/datatype/DurationImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLTdElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xml/XMLEntityDescription.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/util/XSNamedMapImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/Constants.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/events/MouseEventImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dtd/XMLDTDValidatorFilter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/XMLDocumentScannerImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/Util.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xinclude/XInclude11TextReader.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLOptgroupElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLOptgroupElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/SchemaValidatorConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/DefaultValidationErrorHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/DOMErrorImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/DOMErrorImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLPostfieldElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/StAXSchemaParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/ReadOnlyGrammarPool.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/util/DatatypeMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/util/XMLAttributesImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDAttributeTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDComplexTypeTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDAttributeTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDComplexTypeTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDComplexTypeTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/WMLElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/dom/WMLNoopElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/Version.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/util/IntStack.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/io/ASCIIReader.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xs/XSIDCDefinition.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/WMLCardElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/WMLFieldsetElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/AttrImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/WMLAElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/opti/DefaultDocument.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/FilePathToURI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/ObjectFactory.java
* /opt/rena/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLPrevElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/util/HexBin.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/jaxp/datatype/SerializedXMLGregorianCalendar.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/parser/XMLDTDContentModelFilter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/parser/XMLComponentManager.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLBaseFontElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/ddt/XMLDTDLoader.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/DeferredElementDefinitionImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xs/XSElementDeclHelper.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/parser/XMLComponentManager.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLOneventElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/xs/SchemaDVFactoryImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xinclude/XIncludeHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/util/SAXMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/DeferredCommentImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/tab/dom/HTMLTableElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/ElementNSImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/AttributeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/regex/RegexParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/xstax/events/AttributeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/stax/events/EntityDeclarationImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/xpath/XPathException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/AbstractSAXParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/wml/WMLUElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLBigElementImpl.java
Open Source Used In DNAC Platform Arica 7040

* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xni/XMLLocator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xs/XSTypeDefinition.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xs/ItemPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/util/URI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dom/HTMLObjectElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/AnyAtomicDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/opti/NamedNodeMapImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLBuilder.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/WeakReferenceXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xpath/regex/BMPattern.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLLegendElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/XSFacets.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/xs/ElementPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/wml/WMLRefreshElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/ItemPSVI.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/YearMonthDurationDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/AnyAtomicDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/optimizations/XSQName.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/dom/NodeIteratorImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/util/DefaultErrorHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xpath/regex/BMPattern.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/SimpleXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLLegendElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/YearMonthDurationDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/DXSTraversers.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLHeadingElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLHeadingElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/jaxp/validation/SimpleXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/html/dom/HTMLLegendElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/YearMonthDurationDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/dv/xs/Traversers/XSDWildcardTraverser.java
Open Source Used In DNAC Platform Artica 7042
Open Source Used In DNAC Platform Arica  7043

jar/org/apache/xerces/impl/xs/AttributePSVImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/WMLSelectElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/XMLString.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/WMLSelectElement.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/HTMLTableRowElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/traversers/XSDAbstractTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/NamedNodeMapImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xml/serialize/Printer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dtd/XML11DTDProcessor.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/parser/XMLDocumentFilter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLGoElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/facet/Printer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLRefreshElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/ParserForXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/ParserForXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/HTMLTableRowElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/util/XSObjectListImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLGoElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/ParserForXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dtd/XMLDTDDescription.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/PrecisionDecimalDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/PrecisionDecimalDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/XMLPullParserConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/xni/parser/XMLPullParserConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/parser/SecureProcessingConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xml/serialize/Printer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLRefreshElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/traversers/XSAbstractTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xml/serialize/Printer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dtd/XML11DTDProcessor.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dtd/XML11DTDProcessor.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dtd/XMLDTDDescription.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/wml/dom/WMLRefreshElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/ParserForXMLSchema.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/dom/DOMNormalizer.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dv/xx/IntegerDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/jaxp/DocumentBuilderFactoryImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/html/dom/html/HTMLIsIndexElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/dom/DeferredDocumentImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dtd/models/SimpleContentModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dtd/models/DeferredDocumentImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dtd/models/DeferredDOMImplementationImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/LSInputListImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/xs/XSDeclarationPool.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/jaxp/validation/JAXPValidationMessageFormatter.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dtd/models/ContentModel.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
  jar/org/apache/xerces/impl/dtd/impl/DefaultCharacterDataImpl.java
Open Source Used In DNAC Platform Arica 7045
Open Source Used In DNAC Platform Arica  7046

jar/org/apache/xerces/impl/xs/traversers/XSDNotationTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/grammars/XMLSchemaDescription.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/jaxp/validation/DOMResultBuilder.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLAnchorElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/html/dom/HTMLModElementImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/wml/dom/WMLDocumentImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/DeferredElementNSImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/XMLEntityHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/StandardParserConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/RevalidationHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xs/XSValue.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/NodeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/IncludeAwareParserConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/RevalidationHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/io/UCSReader.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/XIncludeAwareParserConfiguration.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/dtd/XML11IDREFDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/dom/NodeImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/impl/dv/dtd/XML11IDREFDatatypeValidator.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/xni/parser/XMLConfigurationException.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/util/AttributesProxy.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/jaxp/validation/ValidatorImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xerces/parsers/XMLGrammarParser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/dtd/XMLSimpleType.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xs/AbstractDateTimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xs/YearDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/xx/SchemaNamespaceSupport.java

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/AbstractDateTimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/YearDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/xx/traversers/XSDKyrefTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/DateDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/DateTimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/traversers/XSDUniqueOrKeyTraverser.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/month/XXMonthDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources.jar/org/apache/xercesimpl/impl/dv/xx/DurationDV.java

Open Source Used In DNAC Platform Arica 7047
jar/org/apache/xerces/impl/dv/xs/MonthDayDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/TimeDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/traversers/XSAttributeChecker.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/YearMonthDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/XSModelGroupImpl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/XSSimpleTypeDecl.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/traversers/XSDHandler.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/dv/xs/DayDV.java
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xs/XMLSchemaValidator.java
No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/XMLSchemaMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/message.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/DOMMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/XPointerMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/DatatypeMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/message_fr.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/SAXMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/msg/XMLSerializerMessages.properties
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-
jar/org/apache/xerces/impl/xpath/regex/message_ja.properties
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

// Unique Particle Attribution

Found in path(s):
* /opt/cola/permits/1257218116_1642790294.82/0/xercesimpl-2-12-0-sources-jar/org/apache/xerces/impl/xs/models/XSDFACM.java

1.1576 is-property 1.0.2 1.0.2
1.1576.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013 Mikola Lysenko

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1577 path-key 2.0.1 2.0.1
1.1577.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1578 is-weakref 1.0.1
1.1578.1 Available under license:
MIT License

Copyright (c) 2020 Inspect JS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1579 mail 1.5.6
1.1579.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally
1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the
jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of...
any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started
running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This
alternative is allowed only for noncommercial distribution and only if you received the program in object code or
executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable
work, complete source code means all the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is normally distributed (in either source or
binary form) with the major components (compiler, kernel, and so on) of the operating system on which the
executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time
to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.1580 kerb-client 1.0.0
1.1580.1 Available under license :

Kerby-kerb Client
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1581 kafka-schema-serializer 6.1.1
1.1581.1 Available under license:

Confluent Community License Agreement
Version 1.0

This Confluent Community License Agreement Version 1.0 (the Agreement) sets forth the terms on which Confluent, Inc. (Confluent) makes available certain software made available by Confluent under this Agreement (the Software). BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO SUCH TERMS AND CONDITIONS, YOU MUST NOT USE THE SOFTWARE. IF YOU ARE RECEIVING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY. Licensee means you, an individual, or
the entity on whose behalf you are receiving the Software.

1. LICENSE GRANT AND CONDITIONS.

1.1 License. Subject to the terms and conditions of this Agreement, Confluent hereby grants to Licensee a non-exclusive, royalty-free, worldwide, non-transferable, non-sublicenseable license during the term of this Agreement to: (a) use the Software; (b) prepare modifications and derivative works of the Software; (c) distribute the Software (including without limitation in source code or object code form); and (d) reproduce copies of the Software (the License). Licensee is not granted the right to, and Licensee shall not, exercise the License for an Excluded Purpose. For purposes of this Agreement, Excluded Purpose means making available any software-as-a-service, platform-as-a-service, infrastructure-as-a-service or other similar online service that competes with Confluent products or services that provide the Software.

1.2 Conditions. In consideration of the License, Licensee’s distribution of the Software is subject to the following conditions:

(a) Licensee must cause any Software modified by Licensee to carry prominent notices stating that Licensee modified the Software.

(b) On each Software copy, Licensee shall reproduce and not remove or alter all Confluent or third party copyright or other proprietary notices contained in the Software, and Licensee must provide the notice below with each copy.

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at http://www.confluent.io/confluent-community-license. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

1.3 Licensee Modifications. Licensee may add its own copyright notices to modifications made by Licensee and may provide additional or different license terms and conditions for use, reproduction, or distribution of Licensee’s modifications. While redistributing the Software or modifications thereof, Licensee may choose to offer, for a fee or free of charge, support, warranty, indemnity, or other obligations. Licensee, and not Confluent, will be responsible for any such obligations.

1.4 No Sublicensing. The License does not include the right to sublicense the Software, however, each recipient to which Licensee provides the Software may exercise the Licenses so long as such recipient agrees to the terms and conditions of this Agreement.

2. TERM AND TERMINATION. This Agreement will continue unless and until
earlier terminated as set forth herein. If Licensee breaches any of its
conditions or obligations under this Agreement, this Agreement will
terminate automatically and the License will terminate automatically and
permanently.

3. INTELLECTUAL PROPERTY. As between the parties, Confluent will retain all
right, title, and interest in the Software, and all intellectual property
rights therein. Confluent hereby reserves all rights not expressly granted
to Licensee in this Agreement. Confluent hereby reserves all rights in its
trademarks and service marks, and no licenses therein are granted in this
Agreement.

4. DISCLAIMER. CONFLUENT HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND
CONDITIONS, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND SPECIFICALLY
DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR
PURPOSE, WITH RESPECT TO THE SOFTWARE.

5. LIMITATION OF LIABILITY. CONFLUENT WILL NOT BE LIABLE FOR ANY DAMAGES OF
ANY KIND, INCLUDING BUT NOT LIMITED TO, LOST PROFITS OR ANY CONSEQUENTIAL,
SPECIAL, INCIDENTAL, INDIRECT, OR DIRECT DAMAGES, HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THE FOREGOING SHALL
APPLY TO THE EXTENT PERMITTED BY APPLICABLE LAW.

6. GENERAL.

6.1 Governing Law. This Agreement will be governed by and interpreted in
accordance with the laws of the state of California, without reference to
its conflict of laws principles. If Licensee is located within the
United States, all disputes arising out of this Agreement are subject to
the exclusive jurisdiction of courts located in Santa Clara County,
California, USA. If Licensee is located outside of the United States,
you dispute, controversy or claim arising out of or relating to this
Agreement will be referred to and finally determined by arbitration in
accordance with the JAMS International Arbitration Rules. The tribunal
will consist of one arbitrator. The place of arbitration will be Palo
Alto, California. The language to be used in the arbitral proceedings
will be English. Judgment upon the award rendered by the arbitrator may
be entered in any court having jurisdiction thereof.

6.2 Assignment. Licensee is not authorized to assign its rights under
this Agreement to any third party. Confluent may freely assign its rights
under this Agreement to any third party.

6.3 Other. This Agreement is the entire agreement between the parties
regarding the subject matter hereof. No amendment or modification of
this Agreement will be valid or binding upon the parties unless made in
writing and signed by the duly authorized representatives of both
parties. In the event that any provision, including without limitation
any condition, of this Agreement is held to be unenforceable, this
Agreement and all licenses and rights granted hereunder will immediately
terminate. Waiver by Confluent of a breach of any provision of this
Agreement or the failure by Confluent to exercise any right hereunder
will not be construed as a waiver of any subsequent breach of that right
or as a waiver of any other right.

Apache Kafka
Copyright 2016 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL
License. The source code of jersey can be found at https://github.com/jersey/jersey/.
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

## Licensing

Jackson core and extension components may licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.
Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel
This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
Apache License, Version 2.0

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 . Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. -->
The following code is added by mdx_elementid.py

It was originally lifted from http://subversion.apache.org/style/site.css *

* Hide class="elementid-permalink", except when an enclosing heading
* has the :hover property.
*

 Apox License  

 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the
copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other
entities that control, are controlled by, or are under common control with
that entity. For the purposes of this definition, "control" means (i) the
power, direct or indirect, to cause the direction or management of such
entity, whether by contract or otherwise, or (ii) ownership of fifty
percent (50%) or more of the outstanding shares, or (iii) beneficial
ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising
permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source,
and configuration files.

"Object" form shall mean any form resulting from mechanical transformation
or translation of a Source form, including but not limited to compiled
object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form,
made available under the License, as indicated by a copyright notice that
is included in or attached to the work (an example is provided in the
Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form,
that is based on (or derived from) the Work and for which the editorial
revisions, annotations, elaborations, or other modifications represent, as
a whole, an original work of authorship. For the purposes of this License,
Derivative Works shall not include works that remain separable from, or
merely link (or bind by name) to the interfaces of, the Work and Derivative
Works thereof.

"Contribution" shall mean any work of authorship, including the original
version of the Work and any modifications or additions to that Work or
Derivative Works thereof, that is intentionally submitted to Licensor for
inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."</p>

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.</p>

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.</p>

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.</p>

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions):

- You must give any other recipients of the Work or Derivative Works a copy of this License; and

- You must cause any modified files to carry prominent notices stating that You changed the files; and

- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and
under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
<div class="container">
  <div class="row">
    <div class="col-sm-1">
    </div>
    <div class="col-sm-2">
      <h5 class="white">Community</h5>
      <ul class="list-unstyled white" role="menu">
        <li><a href="http://community.apache.org/">Overview</a></li>
        <li><a href="/foundation/conferences.html">Conferences</a></li>
        <li><a href="http://community.apache.org/newcomers/">Getting Started</a></li>
        <li><a href="/foundation/how-it-works.html">The Apache Way</a></li>
        <li><a href="/travel/">Travel Assistance</a></li>
        <li><a href="/foundation/getinvolved.html">Get Involved</a></li>
        <li><a href="http://community.apache.org/newbiefaq.html">Community FAQ</a></li>
      </ul>
    </div>
    <div class="col-sm-2">
      <h5 class="white">Innovation</h5>
      <ul class="list-unstyled white" role="menu">
        <li><a href="http://incubator.apache.org/">Incubator</a></li>
        <li><a href="http://labs.apache.org/">Labs</a></li>
        <li><a href="/licenses/">Licensing</a></li>
        <li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
        <li><a href="/foundation/marks/">Trademark Policy</a></li>
        <li><a href="/foundation/contact.html">Contacts</a></li>
      </ul>
    </div>
    <div class="col-sm-2">
      <h5 class="white">Tech Operations</h5>
      <ul class="list-unstyled white" role="menu">
        <li><a href="/dev/">Developer Information</a></li>
        <li><a href="/dev/infrastructure.html">Infrastructure</a></li>
        <li><a href="/security/">Security</a></li>
        <li><a href="http://status.apache.org">Status</a></li>
        <li><a href="/foundation/contact.html">Contacts</a></li>
      </ul>
    </div>
    <div class="col-sm-2">
      <h5 class="white">Press</h5>
      <ul class="list-unstyled white" role="menu">
        <li><a href="/press/">Overview</a></li>
      </ul>
    </div>
  </div>
</div>
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This copy of Jackson JSON processor annotations is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
<head>
<title>301 Moved Permanently</title>
</head>
<body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://opensource.org/licenses/mit-license.php">here</a>.</p>
</body>
</html>

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: schema-registry
Source: https://github.com/confluentinc/schema-registry

Files: *
Copyright: 2018 Confluent, Inc.
License: Apache-2

Files: core/*
Copyright: 2015 Confluent, Inc.
License: Confluent Community License

License: Confluent Community License
Licensed under the Confluent Community License; you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.confluent.io/confluent-community-license

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OF ANY KIND, either express or implied. See the License for the
specific language governing permissions and limitations under the License.

License: Apache-2
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

On Debian systems, the Apache 2.0 license can be found in /usr/share/common-licenses/Apache-2.0.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This distribution has a binary dependency on jersey, which is available under the CDDL License as described below.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL - Version 1.1)
1. Definitions.
1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the
name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.
6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S
NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California. The following commands were used to generate license and notice files. Replace <VERSION> with the Schema Registry version, <SRC_PATH> with the path to the Schema Registry source directory, and <LICENSE_TOOL_PATH> with the path of the license tool.

```
cd <SRC_PATH>
mvn package -DskipTests
mkdir /tmp/jars
mkdir /tmp/overrides
```
Apache Avro
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html lang="en-US">
<head>
<title>GlassFish Dual License Header and License Notice GPL v2 and CDDL</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<style type="text/css">
li {display: block;}
</style>
</head>

<body leftmargin="0" topmargin="0" marginheight="0" marginwidth="0" bgcolor="#ffffff">
<h3>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0</h3>

<ul>
<li><p><b>1. Definitions.</b></p>
<ul>
<li><p><b>1.1. &ldquo;Contributor&rdquo;</b> means each individual or entity that creates or contributes to the creation of Modifications.</p>
</li>
</ul>
</li>

<li><p><b>1.2. &ldquo;Contributor Version&rdquo;</b> means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.</p>
</li>
</ul>
1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. "Modifications" means the Source Code and Executable form of any of the following:

- A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
- B. Any new file that contains any part of the Original Software or previous Modification; or
- C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
<li><p>1.12. **“Source Code”** means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</p></li><li><p>1.13. **“You” (or “Your”)** means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, **“You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</p></li><li><p><b>2. License Grants.</b></p><ul><li><p><b>2.1. The Initial Developer Grant.</b></p><p>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:</p><ul><li><p>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</p></li></ul></li></ul></li>
<li>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
</li>
<li>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
</li>
<li>(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
</li>

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an
unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

</p>
</li>
<li>
<p>
<b>(b)</b> under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1)&nbsp;Modifications made by that Contributor (or portions thereof); and (2)&nbsp;the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
</p>
</li>
<li>
<p>
<b>(c)</b> The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
</p>
</li>
<li>
<p>
<b>(d)</b> Notwithstanding Section 2.2(b) above, no patent license is granted: (1)&nbsp;for any code that Contributor has deleted from the Contributor Version; (2)&nbsp;for infringements caused by: (i)&nbsp;third party modifications of Contributor Version, or (ii)&nbsp;the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3)&nbsp;under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.
</p>
</li>
3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so
only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You:
(a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.
6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software and commercial computer software documentation; as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This
U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.
</p>
</li>

<li>
<p><b>9. MISCELLANEOUS.</b></p>
<p>This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.
</p>
</li>

<li>
<p><b>10. RESPONSIBILITY FOR CLAIMS.</b></p>
<p>As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
</p>
</li>
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or
use pieces of it in new free programs; and that you know you can do these things.
</p>
<p>To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.</p>
<p>For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.</p>
<p>We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.</p>
<p>Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.</p>
<p>Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
</p>

Terms and Conditions for COPYING, DISTRIBUTION AND MODIFICATION

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The
"Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

</p>
</li>
<li>
<p>
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
</p>

</li>
</ul>

<p>
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
</p>

<p>
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
</p>

<p>
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
</p>

<li>
<p>
3. You may copy and distribute the Program (or a work based
on it, under Section 2) in object code or executable form under
the terms of Sections 1 and 2 above provided that you also do one
of the following:
</p>
<ul>
<li>
<p>a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,
</p></li>
<li>
<p>b) Accompany it with a written offer, valid for at least three years,
to give any third party, for a charge no more than your cost of
physically performing source distribution, a complete machine-readable
copy of the corresponding source code, to be distributed under the
terms of Sections 1 and 2 above on a medium customarily used for
software interchange; or,
</p></li>
<li>
<p>c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)
</p></li>
</ul>
<p>The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that
is normally distributed (in either source or binary form) with the
major components (compiler, kernel, and so on) of the operating system
on which the executable runs, unless that component itself accompanies
the executable.
</p>
<p>If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to
copy the source code from the same place counts as distribution of the
source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent
obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

</p>
</li>

<li>
<p>END OF TERMS AND CONDITIONS</p>
</li>

<li>
<p><b>How to Apply These Terms to Your New Programs</b></p>
<p>If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.</p>
</li>

<li>
<p>To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.</p>
</li>

<ul>
<li>
<p>One line to give the program’s name and a brief idea of what it does.</p>
</li>
<li>
<p>Copyright (C) <year> <name of author> </p></li>
<li>
<p>This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.</p>
</li>
<li>
<p>This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.</p>
</li>
</ul>
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words:

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes

...
any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.<p>2. License Grants.</p><blockquote>2.1. The Initial Developer Grant.</blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</blockquote><blockquote>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).</blockquote><blockquote>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes available the Original Software to a third party under the terms of this License.</blockquote><blockquote>(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.</blockquote><blockquote>2.2. Contributor Grant.</blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and</blockquote><blockquote>(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).</blockquote><blockquote>(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes available the Modifications to a third party.</blockquote><blockquote>(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; and (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.</blockquote><p>3. Distribution Obligations.</p><blockquote>3.1. Availability of Source Code.</blockquote><blockquote>Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on
or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULDN'T ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S
NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of &ldquo;commercial computer software&rdquo; (as that term is defined at 48 C.F.R. &sect; 252.227-7014(a)(1)) and &ldquo;commercial computer software documentation&rdquo; as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

The GNU General Public License (GPL) Version 2, June 1991
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
of any warranty; and give any other recipients of the Program a copy of this License along with the Program. </p>
<p>You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee. </p>
<p><strong>2</strong>. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions: </p>
<p><strong>a)</strong> You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change. </p>
<p><strong>b)</strong> You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License. </p>
<p><strong>c)</strong> If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)</p>
<p>These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. </p>
<p>Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program. </p>
<p>In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License. </p>
<p><strong>3</strong>. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following: </p>
<p><strong>a)</strong> Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, </p>
<p><strong>b)</strong> Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, </p>
<p><strong>c)</strong> Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)</p>
<p>The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable. </p>
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and any later version, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM &quot;AS IS&quot; WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the &quot;copyright&quot; line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) &lt;year&gt; &lt;name of author&gt;

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, you and others are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public
License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items—whatever suits your program.</p>
<p>You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:</p>
<blockquote>Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.</blockquote>
<blockquote>signature of Ty Coon, 1 April 1989</blockquote>
Ty Coon, President of Vice</blockquote>
<p>This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.</p>
<li style="background-color:yellow;"><p><strong>"CLASSPATH" EXCEPTION TO THE GPL VERSION 2</strong></p>
<br />
Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."<br />
<br />
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination. <br />
<br />
As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.</p>
</li>
and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"
The project is licensed under the Confluent Community License, except for client libs, which is under the Apache 2.0 license.

See LICENSE file in each subfolder for detailed license agreement.
Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License

Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="Home page of The Apache Software Foundation">
</head>
<title>Licenses</title>

<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"; you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. -->

</head>

<body>

<!-- Navigation -->
<header>

<nav class="navbar navbar-default navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">

<span class="sr-only">Toggle navigation</span>

</button>

</div>

</div>

</nav>

</header>

</body>
The Apache Software Foundation uses various licenses to distribute software and documentation, to accept regular contributions from individuals and corporations, and to accept larger grants of existing software products.

These licenses help us achieve our goal of providing reliable and long-lived software products through collaborative open source software development. In all cases, contributors retain full rights to use their original contributions for any other purpose outside of Apache while providing the ASF and its projects the right to distribute and build upon their work within Apache.

All software produced by The Apache Software Foundation or any of its projects or subjects is licensed according to the terms of the documents listed below.

Apache License, Version 2.0 (current)

The 2.0 version of the Apache License was approved by the ASF in 2004. The goals of this license revision have been to reduce the number of frequently asked questions, to allow the license to be reusable without modification by any project (including non-ASF projects), to allow the license to be included by reference instead of listed in every file, to clarify the license on submission of contributions, to require a patent license on contributions that necessarily infringe the contributor's own patents, and to move comments regarding Apache and other inherited attribution notices to a location outside the license terms (the NOTICE file).

The result is a license that is supposed to be compatible with other open source licenses, while remaining true to the original goals of the Apache Group and supportive of collaborative development across both nonprofit and commercial organizations. The Apache Software Foundation is still trying to determine if this version of the Apache License is compatible with the GPL.

All packages produced by the ASF are implicitly licensed under the Apache License, Version 2.0, unless otherwise explicitly stated. More developer documentation on how to apply the Apache License to your work can be found in *Applying the Apache License, Version 2.0*. 

Apache License, Version 1.1 (historic)

The 1.1 version of the Apache License was approved by the ASF in 2000. The
primary change from the 1.0 license is in the 'advertising clause' (section 3 of the 1.0 license); derived products are no longer required to include attribution in their advertising materials, only in their documentation.</p>
<p>Individual packages licensed under the 1.1 version may have used different wording due to varying requirements for attribution or mark identification, but the binding terms were all the same.</p>

<h3 id="1.0">Apache License, Version 1.0 (historic)</h3>
<p>This is the original Apache License which applies only to older versions of Apache packages (such as version 1.2 of the Web server).</p>

<h3 id="clas">Contributor License Agreements</h3>
<p>The ASF desires that all contributors of ideas, code, or documentation to any Apache projects complete, sign, and submit (via fax or email) an Individual Contributor License Agreement (ICLA). The purpose of this agreement is to clearly define the terms under which intellectual property has been contributed to the ASF and thereby allow us to defend the project should there be a legal dispute regarding the software at some future time. A signed ICLA is required to be on file before an individual is given commit rights to an ASF project.</p>
<p>For a corporation that has assigned employees to work on an Apache project, a Corporate CLA (CCLA) is available for contributing intellectual property via the corporation, that may have been assigned as part of an employment agreement. Note that a Corporate CLA does not remove the need for every developer to sign their own ICLA as an individual, to cover any of their contributions which are not owned by the corporation signing the CCLA.</p>
<p>The ICLA is not tied to any employer you may have, so it is recommended to use one's personal email address in the contact details, rather than an @work address.</p>
<p>Your Full name will be published unless you provide an alternative Public name. For example if your full name is Andrew Bernard Charles Dickens, but you wish to be known as Andrew Dickens, please enter the latter as your Public name.</p>
<p>The email address and other contact details are not published.</p>

<h1 id="grants">Software Grants</h1>
<p>When an individual or corporation decides to donate a body of existing software or documentation to one of the Apache projects, they need to execute a formal Software Grant Agreement (SGA) with the ASF. Typically, this is done after negotiating approval with the ASF Incubator or one of the PMCs, since the ASF will not accept software unless there is a viable community available to support a collaborative project.</p>

<h1 id="submitting">Submitting License Agreements and Grants</h1>
<p>Documents may be submitted by fax or email.</p>
If submitting by email, please fill the form with a PDF viewer, print, sign, scan all pages into a single PDF file, and email the PDF file as an attachment to secretary@apache.org. If possible, send the attachment from the email address in the document. Please send only one document per email.

If you prefer to sign electronically, please fill the form, save it locally (e.g., icla.pdf), and sign the file by preparing a detached PGP signature. For example,

```bash
$ gpg --armor --detach-sign icla.pdf
```

The above will create a file icla.pdf.asc. Send both the file and signature as attachments in the same email to secretary@apache.org. Please send only one document (file plus signature) per email. Please do not submit your public key to Apache. Instead, please upload your public key to pgpkeys.mit.edu.

The files typically are named icla.pdf and icla.pdf.asc for individual agreements; ccla.pdf and ccla.pdf.asc for corporate agreements; software-grant.pdf and software-grant.pdf.asc for grants.

For export restriction information, please consult our ASF Export Classifications page.

For ASF trademark and logo usage information, please consult our ASF Trademark Use Policy page.

For answers to frequently asked licensing questions, please consult our Licensing Frequently Asked Questions page.

--- Footer ---

Community

- Overview
- Conferences
- Summer of Code
- Getting Started
- The Apache Way
- Travel Assistance
This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.

For additional credits (generally to people who reported problems) see CREDITS file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1582 @babel/helper-skip-transparent-expression-wrappers 7.16.0

1.1582.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1583 ms 2.1.3 2.1.3

1.1583.1 Available under license:

The MIT License (MIT)

Copyright (c) 2020 Vercel, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1584 javassist 3.18.2-GA

1.1584.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 2004 Bill Burke. All Rights Reserved.
 *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 */

Found in path(s):
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/ArrayMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/DoubleMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/EnumMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/MemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/Annotation.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/CharMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/ShortMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/MemberValueVisitor.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-
jar/javassist/bytecode/annotation/ClassMemberValue.java
No license file was found, but licenses were detected in source scan.

*/
* Javassist, a Java-bytecode translator toolkit.
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
*
* The contents of this file are subject to the Mozilla Public License Version
* 1.1 (the "License"); you may not use this file except in compliance with
* the License. Alternatively, the contents of this file may be used under
* the terms of the GNU Lesser General Public License Version 2.1 or later,
* or the Apache License Version 2.0.
*
* Software distributed under the License is distributed on an "AS IS" basis,
* WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
* for the specific language governing rights and limitations under the
* License.
*/

Found in path(s):
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/StringMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/BooleanMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/LongMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/IntegerMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/AnnotationMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/ByteMemberValue.java
* /opt/cola/permits/1110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/annotation/FloatMemberValue.java

Open Source Used In DNAC Platform Arica  7146
jar/javassist/bytecode/LocalVariableAttribute.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/ClassFilePrinter.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/tools/framedump.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/stackmap/Tracer.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/CtMethod.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/scopedpool/ScopedClassPoolRepositoryImpl.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/FieldInfo.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/Parser.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/analysis/Util.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/tools/web/BadHttpRequest.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/CtNewMethod.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/Descriptor.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/ast/BinExpr.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/NoFieldException.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/analysis/IntQueue.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/util/proxy/RuntimeSupport.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/ast/CondExpr.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/tools/rmi/ObjectImporter.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/ClassPoolTail.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/ast/Variable.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/Opcode.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/tools/reflect/CannotReflectException.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/util/HotSwapper.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/SerialVersionUID.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/util/CodeConverter.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/analysis/Executor.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/bytecode/CodeAnalyzer.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/CodeAnalyzer.java
* /opt/cola/permits/110923499_1606873622.06/0/javassist-3-18-2-ga-sources-5-jar/javassist/compiler/ast/NewExpr.java
1.1585 mocha 7.2.0
1.1585.1 Available under license:
(The MIT License)

Copyright (c) 2011-2020 OpenJS Foundation and contributors, https://openjsf.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1586 reactor-core 3.0.7.RELEASE

1.1586.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2011-2016 Pivotal Software Inc, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2011-2017 Pivotal Software Inc, All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/BlockingMonoSubscriber.java
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoFilterWhen.java
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/scheduler/DisposableContainer.java
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/scheduler/EmptyDisposableContainer.java
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/scheduler/DisposableRunnable.java
* /opt/ws_local PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/scheduler/ScheduledRunnable.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2011-2016 Pivotal Software Inc, All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/concurrent/OpenHashSet.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/concurrent/QueueSupplier.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/scheduler/ParallelScheduler.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/function/Tuples.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/scheduler/ExecutorScheduler.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/scheduler/ElasticScheduler.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/BlockingFirstSubscriber.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/SynchronousSink.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/SynchronizingSink.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/scheduler/ExecutorServiceScheduler.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/function/Tuple6.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/function/Tuple4.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/util/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/PackageInfo.java
/*
 * Copyright (c) 2011-2017 Pivotal Software Inc, All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */

No license file was found, but licenses were detected in source scan.
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxBufferBoundary.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/MonoPublishOn.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/ParallelFluxHide.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/BlockingSink.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxSkipUntil.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxRepeat.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/ConnectableFluxOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/MonoMaterialize.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxPublishMulticast.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/MonoRunnable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/ParallelArraySource.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxZip.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/adapter/JdkFlowAdapter.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/LambdaSubscriber.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxRetryWhen.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/MonoUsing.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/FluxPeekStateful.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/MonoSingle.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-
  jar/reactor/core/publisher/LambdaMonoSubscriber.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoNever.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxConcatIterable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ConnectableFlux.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxRepeatWhen.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxOnBackpressureLatest.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/InnerConsumer.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoFilterFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoCallableOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Producer.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxBatch.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoHasElements.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxBufferPredicate.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/Loopback.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoReduce.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/SignalPeek.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxBufferTimeOrSize.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxTakeLastOne.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoHandleFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSubscribeOnCallable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoLogFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxReCount.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxWindowTimeOrSize.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSwitchMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxMapSignal.java
Open Source Used In DNAC Platform Arica 7162

* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelCollect.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/package-info.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxFlatMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoElapsed.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoWhen.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoLog.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxFlattenIterable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/TopicProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxArray.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxMerge.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSkipUntilOther.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Receiver.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSampleFirst.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSampleTimeout.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSample.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxAwaitOnSubscribe.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxCallableOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxNever.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxEmpty.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDetach.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoMapFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDistinct.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelMergeSort.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/EventLoopProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoUntilOther.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxWithLatestFrom.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxConcatArray.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoSink.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxFilter.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelConcatMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDistinctUntilChanged.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxLog.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelFlatMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoRetry.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxBufferWhen.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxUsing.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxElapsed.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxJust.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSource.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/InnerOperator.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/DirectProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSkipLast.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoRetryWhen.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSubscribeOn.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Operators.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxWindowBoundary.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxWindowOnCancel.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoPeekFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoStreamCollector.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/EmitterProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoDelay.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxRange.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoSubscribeOn.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelSource.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/Sasurable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoPeek.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxGroupBy.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoAll.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDoFinally.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/BlockingSingleSubscriber.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxBuffer.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoDoFinally.java
* /opt.ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxOnBackpressureDrop.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/SignalType.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoCompletionStage.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/Trackable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxTakeUntilOther.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoCollect.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxHide.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/MultiProducer.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/SignalPeekStateful.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Flux.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoSubscribeOnValue.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoError.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Mono.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoAwaitOnSubscribe.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoDelaySubscription.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/DrainUtils.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoCreate.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxGenerate.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxCreate.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoDetach.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxZipIterable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxWindow.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelReduceSeed.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelFluxOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDistinctFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/InnerProducer.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoSwitchIfEmpty.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoSource.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoNext.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxPeek.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxMergeSequential.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/Fuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxSubscribeOnValue.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxOnErrorResume.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxJoin.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoJust.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxJoinable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxIterable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxConcatMap.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxPublish.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxTakeWhile.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/BaseSubscriber.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/UnicastProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ParallelPeek.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/MultiReceiver.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxTakeFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/Schedulers.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxMaterialize.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxHandle.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoRetryPredicate.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoHandle.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxHandleFuseable.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxCombineLatest.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/MonoCancelOn.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/ReplayProcessor.java
* /opt/ws_local/PERMITS_SQL/1069088736_1594483108.64/0/reactor-core-3-0-7-release-sources-jar/reactor/core/publisher/FluxDoFinallyFuseable.java
1.1587 serve-favicon 2.5.0 2.5.0

1.1587.1 Available under license:

(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1588 deep-sort-object 1.0.2 1.0.2

1.1588.1 Available under license:

Copyright (c) 2014 IndigoUnited

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished
to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1589 parser-core 5.7

1.1589.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Apache HTTPD & NGINX Access log parsing made easy
 * Copyright (C) 2011-2019 Niels Basjes
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
1.1590 compressible 2.0.18 2.0.18

1.1590.1 Available under license:
(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Jeremiah Senkpiel <fishrock123@rocketmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
 TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1591 follow-redirects 1.14.5 1.14.5

1.1591.1 Available under license:

Copyright 2014-present Olivier Lalonde <olalonde@gmail.com>, James Talmage <james@talmage.io>, Ruben Verborgh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1592 junit-jupiter-engine 5.7.0

1.1592.1 Available under license:

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

### 1. Definitions

Contribution means:

* **(a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **(b)** in the case of each subsequent Contributor:
* **(i)** changes to the Program, and
* **(ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.
Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

### 2. Grant of Rights

**a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

**b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

**c)** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute
the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

**(d)** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

**(e)** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

### 3. Requirements

**(3.1)** If a Contributor Distributes the Program in any form, then:

* **(a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **(b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **(i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **(ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **(iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **(iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

**(3.2)** When the Program is Distributed as Source Code:

* **(a)** it must be made available under this Agreement, or if the Program *(i)* is combined with other material in a separate file or files made available under a Secondary License, and *(ii)* the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **(b)** a copy of this Agreement must be included with each copy of the Program.

**(3.3)** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

### 4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor.
Open Source Used In DNAC Platform Arica 7174

(Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

### 5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

### 6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### 7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

#### Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.
1.1593 babel-preset-es2015 6.24.1
1.1593.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Jason Miller

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1594 faker 4.1.0 4.1.0
1.1594.1 Available under license:

faker.js - Copyright (c) 2010-2015 Matthew Bergman & Marak Squires
http://github.com/marak/faker.js/

faker.js was inspired by and has used data definitions from:

* https://github.com/stumpy/faker/ - Copyright (c) 2007-2010 Benjamin Curtis
* http://search.cpan.org/~jasonk/Data-Faker-0.07/ - Copyright 2004-2005 by Jason Kohles

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1595 array-unique 0.3.2 0.3.2

1.1595.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
**1.1596 tmp 0.1.0 0.1.0**

1.1596.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 KARASZI Istvn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.1597 bonecp 0.8.0.RELEASE**

1.1597.1 Available under license:
No license file was found, but licenses were detected in source scan.

```*/
* Copyright 2010 Wallace Wadge
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/```
Found in path(s):
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/ConnectionHandle.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/StatementHandle.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/proxy/package-info.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/ConnectionPartition.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/BoneCPConfig.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/package-info.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/StatisticsMBean.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/DefaultConnectionStrategy.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/ConnectionStrategy.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/proxy/CallableStatementProxy.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/UsernamePassword.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/hooks/CloseThreadMonitor.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/hooks/AcquireFailConfig.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/PoolUtil.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/IStatementCache.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/PreparedStatement.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/AbstractConnectionStrategy.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/MemorizeTransactionProxy.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/hooks/AbstractConnectionHook.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/CloseThreadMonitor.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/hooks/AcquireFailConfig.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/PreparedStatement.java
* /opt/cola/permits/1128404392_1611918002.64/0/bonecp-0-8-0-release-sources-4-jar/com/jolbox/bonecp/hooks/AbstractConnectionHook.java

Open Source Used In DNAC Platform Arica  7183
1.1598 lodash-reject 4.6.0 4.6.0

1.1598.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

**1.1599 is-relative 1.0.0 1.0.0**

**1.1599.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1600 cookies 0.7.3 0.7.3
1.1600.1 Available under license :
(The MIT License)

Copyright (c) 2014 Jed Schmidt, http://jed.is/
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1601 to-utf8 0.0.1 0.0.1
1.1601.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1602 posix-character-classes 0.1.1 0.1.1
1.1602.1 Available under license :
The MIT License (MIT)

Copyright (c) 2016-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1603 commons-math 3.1.1
1.1603.1 Available under license :
Apache Commons Math

Copyright 2001-2012 The Apache Software Foundation
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

===============================================================================

The BracketFinder (package org.apache.commons.math3.optimization.univariate)
and PowellOptimizer (package org.apache.commons.math3.optimization.general)
classes are based on the Python code in module "optimize.py" (version 0.5)
developed by Travis E. Oliphant for the SciPy library (http://www.scipy.org/)

===============================================================================

The LinearConstraint, LinearObjectiveFunction, LinearOptimizer,
RelationShip, SimplexSolver and SimplexTableau classes in package
org.apache.commons.math3.optimization.linear include software developed by
Benjamin McCann (http://www.bennccann.com) and distributed with
the following copyright: Copyright 2009 Google Inc.

===============================================================================

This product includes software developed by the
University of Chicago, as Operator of Argonne National
Laboratory.
The LevenbergMarquardtOptimizer class in package
org.apache.commons.math3.optimization.general includes software
translated from the lmder, lmpar and qrsolv Fortran routines
from the Minpack package
Minpack Copyright Notice (1999) University of Chicago. All rights reserved

===============================================================================

The GraggBulirschStoerIntegrator class in package
org.apache.commons.math3.ode.nonstiff includes software translated
from the odex Fortran routine developed by E. Hairer and G. Wanner.
Original source copyright:
Copyright (c) 2004, Ernst Hairer

===============================================================================

The EigenDecompositionImpl class in package
org.apache.commons.math3.linear includes software translated
from some LAPACK Fortran routines. Original source copyright:
Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

===============================================================================

The MersenneTwister class in package org.apache.commons.math3.random
includes software translated from the 2002-01-26 version of
the Mersenne-Twister generator written in C by Makoto Matsumoto and Takuji
Nishimura. Original source copyright:
Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,
All rights reserved
The LocalizedFormatsTest class in the unit tests is an adapted version of
the OrekitMessagesTest class from the orekit library distributed under the
terms of the Apache 2 licence. Original source copyright:
Copyright 2010 CS Systèmes d'Information

The HermiteInterpolator class and its corresponding test have been imported from
the orekit library distributed under the terms of the Apache 2 licence. Original
source copyright:
Copyright 2010-2012 CS Systèmes d'Information

The creation of the package "o.a.c.m.analysis.integration.gauss" was inspired
by an original code donated by Sébastien Brisard.

The complete text of licenses and disclaimers associated with the the original
sources enumerated above at the time of code translation are in the LICENSE.txt
file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersedes or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS MATH DERIVATIVE WORKS:

The Apache commons-math library includes a number of subcomponents whose implementation is derived from original sources written in C or Fortran. License terms of the original sources are reproduced below.

===============================================================================
For the lmder, lmpar and qrsolv Fortran routine from minpack and translated in the LevenbergMarquardtOptimizer class in package org.apache.commons.math3.optimization.general
Original source copyright and license statement:

Minpack Copyright Notice (1999) University of Chicago. All rights reserved
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE,
EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

===============================================================================

Copyright and license statement for the odex Fortran routine developed by E. Hairer and G. Wanner and translated in GraggBulirschStoerIntegrator class in package org.apache.commons.math3.ode.nonstiff:

Copyright (c) 2004, Ernst Hairer

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

Copyright and license statement for the original lapack fortran routines translated in EigenDecompositionImpl class in package org.apache.commons.math3.linear:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

$COPYRIGHT$

Additional copyrights may follow

$HEADER$

Redistribution and use in source and binary forms, with or without
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.

Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright and license statement for the original Mersenne twister C routines translated in MersenneTwister class in package org.apache.commons.math3.random:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

The class "org.apache.commons.math3.exception.util.LocalizedFormatsTest" is an adapted version of "OrekitMessagesTest" test class for the Orekit library. The "org.apache.commons.math3.analysis.interpolation.HermiteInterpolator" has been imported from the Orekit space flight dynamics library.

Th Orekit library is described at: https://www.orekit.org/forge/projects/orekit
The original files are distributed under the terms of the Apache 2 license which is: Copyright 2010 CS Communication & Systèmes

1.1604 streamifier 0.1.1 0.1.1
1.1604.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Gabriel Llamas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1605 parquet-hadoop 1.10.0

1.1605.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

-----------------------------------------------------------------------------

This product includes code from Twitter's ElephantBird project.

* parquet-hadoop's UnmaterializableRecordCounter.java includes code from ElephantBird's LzoRecordReader.java
1.1606 metrics-jmx 4.0.2

1.1607 accepts 1.3.7 1.3.7

1.1607.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1608 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava

1.1608.1 Available under license:
Apache License

Version 2.0, January 2004
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce,
prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.1609 batik-util 1.13
1.1609.1 Available under license:

No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/resources/Messages.properties
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/resources/XMLResourceDescriptor.properties
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/io/resources/Messages.properties

No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/Base64DecodeStream.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/UTF16Decoder.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/ParsedURLData.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/ParsedURL.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/ParsedURLDataProtocolHandler.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/XBLConstants.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/ASCIIDecoder.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/StringNormalizedReader.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/CleanerThread.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/MimeTypeConstants.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/StringNormalizingReader.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/DoublyIndexedTable.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/io/StringDecoder.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/ParsedURLJarProtocolHandler.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
  jar/org/apache/batik/util/DOMConstants.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/DoublyIndexedSet.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/ParsedURLDefaultProtocolHandler.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/PreferenceManager.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/RunnableQueue.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/BatikSecurityManager.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/ClassFileUtilities.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/io/UTF8Decoder.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/io/NormalizingReader.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/RunnableQueue.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/PreferenceManager.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/ApplicationSecurityEnforcer.java
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/DoublyIndexedSet.java

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1135872869_1613620994.25/0/batik-util-1-13-sources-1-jar/org/apache/batik/util/RunnableQueue.java

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

1.1610 mime 1.6.0 1.6.0

1.1610.1 Available under license:
The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1611 type-is 1.6.16 1.6.16
1.1611.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1612 acorn 5.7.4
1.1612.1 Available under license:

Copyright (C) 2012-2018 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.1613 jsoup 1.9.2

1.1613.1 Available under license:

The MIT License

2009-2016, Jonathan Hedley <jonathan@hedley.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1614 extend-shallow 2.0.1

1.1614.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1615 jetty-util 9.3.27.v20190418

1.1616 word-wrap 1.2.3 1.2.3

1.1616.1 Available under license:

   The MIT License (MIT)

   Copyright (c) 2014-2017, Jon Schlinkert

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1617 async 2.6.0 2.6.0

1.1617.1 Available under license:

   Copyright (c) 2010-2017 Caolan McMahon

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1618 hamcrest-library 1.3
1.1618.1 Available under license:
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.619 lucene-queryparser 7.4.0

1.619.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications...
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from unicode conversion examples available at
http://www.unicode.org/Public/PROGRAMS/CVTUTF. Here is the copyright
from those sources:

/*
 * Copyright 2001-2004 Unicode, Inc.
 *
 * Disclaimer
 *
 * This source code is provided as is by Unicode, Inc. No claims are
 * made as to fitness for any particular purpose. No warranties of any
 * kind are expressed or implied. The recipient agrees to determine
 * applicability of information provided. If this file has been
 * purchased on magnetic or optical media from Unicode, Inc., the
 * sole remedy for any claim will be exchange of defective media
 * within 90 days of receipt.
 *
 * Limitations on Rights to Redistribute This Code
 *
 * Unicode, Inc. hereby grants the right to freely use the information
 * supplied in this file in the creation of products supporting the
 * Unicode Standard, and to make copies of this file in any form
 * for internal or external distribution as long as this notice
 * remains attached.
 */

Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was
derived from Python 2.4.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was derived from Python 3.1.2 sources available at http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in core/src/java/org/apache/lucene/util/automaton was derived from Brics automaton sources available at www.brics.dk/automaton/. Here is the copyright from those sources:

/*
 * Copyright (c) 2001-2009 Anders Moeller
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 *    derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

The levenshtein automata tables in core/src/java/org/apache/lucene/util/automaton were automatically generated with the moman/finenight FSA package.

Here is the copyright for those sources:

# Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
# Permission is hereby granted, free of charge, to any person
# obtaining a copy of this software and associated documentation
# files (the "Software"), to deal in the Software without
# restriction, including without limitation the rights to use,
# copy, modify, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following
# conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
# OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
# OTHER DEALINGS IN THE SOFTWARE.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
* Copyright (C) 1999-2010, International Business Machines
* Corporation and others. All Rights Reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, and/or sell copies of the
* Software, and to permit persons to whom the Software is furnished to do so,
* provided that the above copyright notice(s) and this permission notice appear
* in all copies of the Software and that both the above copyright notice(s) and
* this permission notice appear in supporting documentation.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
* IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
* LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
* ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
* IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* Except as contained in this notice, the name of a copyright holder shall not
The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
- this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
- documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holders nor the names of its contributors
- may be used to endorse or promote products derived from this software
- without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

3. The names "Center for Intelligent Information Retrieval" and
"University of Massachusetts" must not be used to endorse or promote products
derived from this software without prior written permission. To obtain
permission, contact info@ciir.cs.umass.edu.

THIS SOFTWARE IS PROVIDED BY UNIVERSITY OF MASSACHUSETTS AND OTHER CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

The dictionary comes from Morfologik project. Morfologik uses data from Polish ispell/myspell dictionary hosted at http://www.sjp.pl/słownik/en/ and is licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike. The part-of-speech tags were added in Morfologik project and are not found in the data from sjp.pl. The tagset is similar to IPI PAN tagset.

---

The following license applies to the Morfeusz project, used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/

Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski, Marcin Woliski, Robert Woosz

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY COPYRIGHT HOLDERS AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects,
including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license
and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such
as the Unicode Character Database. See http://unicode.org/copyright.html for more
details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is
BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were
automatically generated with the moman/finenight FSA library, created by
Jean-Philippe Barrette-LaPierre. This library is available under an MIT license,
see http://sites.google.com/site/rrettesite/moman and
http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from
the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
g Package (jaspell): http://jaspell.sourceforge.net/
License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in
analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from http://snowball.tartarus.org/

The KStem stemmer in analysis/common/src/org/apache/lucene/analysis/en was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst) under the BSD-license.


The German,Spanish,Finnish,French,Hungarian,Italian,Portuguese,Russian and Swedish light stemmers (common) are based on BSD-licensed reference implementations created by Jacques Savoy and Ljiljana Dolamic. These files reside in:
analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/es/SpanishMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fi/FinnishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/it/ItalianMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil, and Edmond Nolan.

The Polish analyzer (stempel) comes with a default stopword list that is BSD-licensed created by the Carrot2 project. The file resides in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt.
See http://project.carrot2.org/license.html.

The SmartChineseAnalyzer source code (smartcn) was provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/) is derived from Unicode data such as the Unicode Character Database. See http://unicode.org/copyright.html for more details.
The Morfologik analyzer (morfologik) includes BSD-licensed software
developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary
(http://www.sjp.pl/słownik/en/) licenced on the terms of (inter alia)
LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP)
(http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original
source code for this can be found at http://www.eclipse.org/jetty/downloads.php

===========================================================================
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

===========================================================================
mecab-ipadic-2.7.0-20070801 Notice
===========================================================================

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear on, or be attached to, the Program, which is distributed substantially in the same form as set out herein and that such intended distribution, if actually made, will neither violate or otherwise contravene any of the laws and regulations of the countries having jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to have agreed and understood, that there is no warranty whatsoever for the program and, accordingly, the entire risk arising from or otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other organization that participated in or was otherwise related to the development of the program and their respective officials, directors, officers and other employees shall be held liable for any and all damages, including, without limitation, general, special, incidental and consequential damages, arising out of or otherwise in connection with the use or inability to use the program or any product, material or result produced or otherwise obtained by using the program, regardless of whether they have been advised of, or otherwise had knowledge of, the possibility of such damages at any time during the project or thereafter. Each user will be deemed to have agreed to the foregoing by his or her commencement of use of the program. The term "use" as used herein includes, but is not limited to, the use, modification, copying and distribution of the program and the production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.
Nori Korean Morphological Analyzer - Apache Lucene Integration

This software includes a binary and/or source version of data from mecab-ko-dic-2.0.3-20170922 which can be obtained from https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz

1.1620 model-mapper 1.1.0

1.1621 kafka-raft 2.8.0
1.1621.1 Available under license:

Apache Kafka
Copyright 2021 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

The streams-scala (streams(streams-scala) module was donated by Lightbend and the original code was copyrighted by them:
Copyright (C) 2018 Lightbend Inc. <https://www.lightbend.com>
Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:
clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java
Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1622 bl 1.2.3 1.2.3

1.1622.1 Available under license:

The MIT License (MIT)

=================================

Copyright (c) 2013-2016 bl contributors

*bl contributors listed at <https://github.com/rvagg/bl#contributors>*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1623 core-js-compat 3.19.1

1.1623.1 Available under license:

Copyright (c) 2014-2021 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.1624 jersey-container-jetty-http 2.25.1

1.1624.1 Available under license:

Found license 'General Public License 2.0' in

```
* Copyright (c) 2011-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License
```

```
# Copyright (c) 2013-2014 Oracle and/or its affiliates. All rights reserved. # The contents of this file are subject to the terms of either the GNU # General Public License Version 2 only ("GPL") or the Common Development # and Distribution License("CDDL") (collectively, the "License"). You # may not use this file except in compliance with the License. You can # Oracle designates this particular file as subject to the "Classpath" # exception as provided by Oracle in the GPL Version 2 section of the License
```

```
# Copyright (c) 2013-2015 Oracle and/or its affiliates. All rights reserved. # The contents of this file are subject to the terms of either the GNU # General Public License Version 2 only ("GPL") or the Common Development # and Distribution License("CDDL") (collectively, the "License"). You # may not use this file except in compliance with the License. You can # Oracle designates this particular file as subject to the "Classpath" # exception as provided by Oracle in the GPL Version 2 section of the License
```

1.1625 esutils 2.0.3

1.1625.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
```
1.1626 glob 7.1.6

1.1626.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## Glob Logo

Glob's logo created by Tanya Brassie <http://tanyabrassie.com/>, licensed under a Creative Commons Attribution-ShareAlike 4.0 International License https://creativecommons.org/licenses/by-sa/4.0/

1.1627 tomcat 8.5.46

1.1627.1 Available under license:
Apache Tomcat

Copyright 1999-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1628 generate-schema 2.6.0 2.6.0

1.1628.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Nijiko Yonskai

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1629 fd-slicer 1.1.0 1.1.0

1.1629.1 Available under license :

Copyright (c) 2014 Andrew Kelley

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1630 expand-brackets 0.1.5
1.1630.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1631 commons-collections 3.2.2
1.1631.1 Available under license:

Apache Commons Collections
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1632 babel-polyfill 6.26.0

1.1633 monorepo-symlink-test 0.0.0
1.1633.1 Available under license:
MIT License

Copyright (c) 2012 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1634 jersey-mvc-freemarker 2.25.1
1.1634.1 Available under license:
Found license 'General Public License 2.0' in /* Copyright (c) 2015 Oracle and/or its affiliates. All rights reserved. */
The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'
Found license 'General Public License 2.0' in /* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. */
The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL")
or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in * Copyright (c) 2013-2015 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License

1.1635 websocket-client 9.3.27.v20190418

1.1636 sindresorhus-is 0.14.0
1.1636.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1637 base16 1.0.0 1.0.0
1.1637.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1638 javassist 3.27.0-GA

1.1638.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 *
 * javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
 * *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 * *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 * */

Found in path(s):
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/rmi/Proxy.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Visitor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Member.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/CannotInvokeException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/CtNewWrappedConstructor.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/LoaderClassPath.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/LoaderClassPath.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/LoaderClassPath.java
Open Source Used In DNAC Platform Arica 7251

jar/javassist/tools/reflect/CannotCreateException.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/EnclosingMethodAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/BinExpr.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/SourceFileAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/TypeChecker.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/FieldDecl.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/ASTree.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Stmnt.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/StackMap.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/NestHostAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/compiler/ast/Variable.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/ClassMetaobject.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Metalevel.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/bytecode/AccessFlag.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/ClassFilePrinter.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Instanceof.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/ConstantAttribute.java
* /opt/cola/permits/1135872664_1613621010.52/0/javassist-3-27-0-ga-sources-1-jar/javassist/tools/reflect/Desc.java
1.1639 drill-protocol 1.19.0

1.1639.1 Available under license:

Drill: Protocol
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation.
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1640 json-path 0.11.2 0.11.2

1.1640.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1.1641 hbase-prefix-tree 2.0.0-alpha4

1.1641.1 Available under license:

Apache HBase - Prefix Tree
Copyright 2007-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1642 commander 2.20.3

1.1642.1 Available under license :
(The MIT License)

Copyright (c) 2011 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
1.1643 jersey 2.32

1.1643.1 Available under license:

# Notice for Jersey Json Jackson module
This content is produced and maintained by the Eclipse Jersey project.

* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Jackson JAX-RS Providers version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.
/*
 * Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.
 * 
 * This program and the accompanying materials are made available under the 
 * terms of the Eclipse Public License v. 2.0, which is available at 
 * 
 * This Source Code may also be made available under the following Secondary 
 * Licenses when the conditions for such availability set forth in the 
 * Eclipse Public License v. 2.0 are satisfied: GNU General Public License, 
 * version 2 with the GNU Classpath Exception, which is available at 
 * 
 * SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0
 */

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Public License v. 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made
available under the following Secondary Licenses when the conditions for such
availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU
General Public License, version 2 with the GNU Classpath Exception which is

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

---

Open Source Used In DNAC Platform Arica  7268
Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation
jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
/*
* Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.
*
* This program and the accompanying materials are made available under the
* terms of the Eclipse Distribution License v. 1.0, which is available at
*
* SPDX-License-Identifier: BSD-3-Clause
*/
# Notice for Jersey Core Server module
This content is produced and maintained by the Eclipse Jersey project.
* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright: (c) 2000-2011 INRIA, France Telecom. All rights reserved.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/

# Notice for Jersey Core Common module
This content is produced and maintained by the Eclipse Jersey project.
* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright: (C) 2009 The Guava Authors

JSR-166 Extension - JEP 266
* License: Creative Commons 1.0 (CC0)
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
* Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.
## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at https://www.eclipse.org/org/documents/edl-v10.php.

SPDX-License-Identifier: BSD-3-Clause

## Source Code

The project maintains the following source code repositories:

* [https://github.com/eclipse-ee4j/jersey/examples](https://github.com/eclipse-ee4j/jersey/examples)

## Third-party Content

Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.9.9
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 7.2
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf.

Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the
Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in 
Exhibit A of this Agreement, then the Program may be made available 
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of 
the Program.

3.3 Contributors may not remove or alter any copyright, patent, 
trademark, attribution notices, disclaimers of warranty, or limitations 
of liability ("notices") contained within the Program from any copy of 
the Program which they Distribute, provided that Contributors may add 
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities 
with respect to end users, business partners and the like. While this 
license is intended to facilitate the commercial use of the Program, 
the Contributor who includes the Program in a commercial product 
offering should do so in a manner which does not create potential 
liability for other Contributors. Therefore, if a Contributor includes 
the Program in a commercial product offering, such Contributor 
("Commercial Contributor") hereby agrees to defend and indemnify every 
other Contributor ("Indemnified Contributor") against any losses, 
damages and costs (collectively "Losses") arising from claims, lawsuits 
and other legal actions brought by a third party against the Indemnified 
Contributor to the extent caused by the acts or omissions of such 
Commercial Contributor in connection with its distribution of the Program 
in a commercial product offering. The obligations in this section do not 
apply to any claims or Losses relating to any actual or alleged 
intellectual property infringement. In order to qualify, an Indemnified 
Contributor must: a) promptly notify the Commercial Contributor in 
writing of such claim, and b) allow the Commercial Contributor to control, 
and cooperate with the Commercial Contributor in, the defense and any 
related settlement negotiations. The Indemnified Contributor may 
participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial 
product offering, Product X. That Contributor is then a Commercial 
Contributor. If that Commercial Contributor then makes performance 
claims, or offers warranties related to Product X, those performance 
claims and warranties are such Commercial Contributor's responsibility 
alone. Under this section, the Commercial Contributor would have to 
defend claims against the other Contributors related to those performance 
claims and warranties, and if a court requires any other Contributor to 
pay any damages as a result, the Commercial Contributor must pay 
those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses...
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Notice for Jersey Bean Validation module
This content is produced and maintained by the Eclipse Jersey project.
* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

1.1644 metrics 0.1.21 0.1.21

1.1644.1 Available under license:

The MIT License (MIT)
Copyright (c) 2012 Mike Ihbe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1645 batik-squiggle-ext 1.13
1.1645.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the

Open Source Used In DNAC Platform Arica 7291
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1646 sqlite 3.25.3
1.1646.1 Available under license :

The author disclaims copyright to this source code. In place of
a legal notice, here is a blessing:

May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.

1.1647 babel-generator 6.26.1 6.26.1
1.1647.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.1648 remark 1.0.0
1.1648.1 Available under license :

# The MIT License (MIT)

Copyright (c) 2018

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1649 @babel/helper-plugin-utils 7.14.5
1.1649.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1650 babel-types 6.26.0 6.26.0
1.1650.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1651 js-yaml 3.13.1
1.1651.1 Available under license :

(The MIT License)

Copyright (C) 2011-2015 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1652 lodash-assignin 4.2.0 4.2.0

1.1652.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1653 strip-ansi 4.0.0 4.0.0

1.1653.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1654 cache-base 1.0.1 1.0.1
1.1654.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1655 @babel/plugin-transform-block-scoped-functions 7.16.0

1.1655.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1656 groovy-macro 3.0.8

1.1656.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/transform/MacroMethodsCache.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/transform/MacroClassTransformation.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/transform/MacroCallTransformingVisitor.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/matcher/internal/MatchingConstraintsBuilder.groovy
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/runtime/MacroStub.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/methods/MacroGroovyMethods.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/matcher/MatcherUtils.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/runtime/MacroStub.java
* /opt/cola/permits/1261640735_1643305293.14/0/groovy-macro-3-0-8-sources-jar/org/codehaus/groovy/macro/methods/MacroGroovyMethods.java
No license file was found, but licenses were detected in source scan.

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
#      http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

#global transformation for macro support
org.codehaus.groovy.macro.transform.MacroTransformation
org.codehaus.groovy.macro.transform.MacroClassTransformation
1.1657 cron-parser 2.7.3 2.7.3

1.1657.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016 Harri Siirak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1658 bindings 1.2.1 1.2.1

1.1658.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1659 eventemitter2 0.4.14 0.4.14
1.1659.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*!
* EventEmitter2
* https://github.com/hij1nx/EventEmitter2
*
* Copyright (c) 2013 hij1nx
* Licensed under the MIT license.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1050207751_1592830586.79/0/eventemitter2-0-4-14-3-tgz/package/lib/eventemitter2.js
No license file was found, but licenses were detected in source scan.

(The MIT License)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal to use, copy, modify, merge, publish, distribute, sublicense, and/or sell to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1050207751_1592830586.79/0/eventemitter2-0-4-14-3-tgz/package/README.md

1.1660 shebang-regex 3.0.0
1.1660.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1661 rimraf 2.4.5 2.4.5
1.1661.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1662 retrofit 2.8.1
1.1662.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
No license file was found, but licenses were detected in source scan.

---

No license file was found, but licenses were detected in source scan.

---

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License";
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Invocation.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/KotlinExtensions.kt
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/HeaderMap.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/CompletableFutureCallAdapterFactory.java

*/

/*
 * Copyright (C) 2016 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License";
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Invocation.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/CompletableFutureCallAdapterFactory.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/HeaderMap.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/HttpException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/ServiceMethod.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Url.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/BuiltInConverters.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Response.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/OkHttpCall.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Call.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/CallAdapter.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/ServiceMethod.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/CallAdapter.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/RequestFactory.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/ParameterHandler.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/OPTIONS.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/FieldMap.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/PartMap.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Streaming.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/QueryMap.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (C) 2019 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/SkipCallbackExecutor.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Tag.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/SkipCallbackExecutorImpl.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (C) 2013 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Open Source Used In DNAC Platform Arica 7308

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Path.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/POST.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Multipart.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Part.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/FORMUrlEncoded.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/HTTP.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Header.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/http/Field.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Platform.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Http.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Headers.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/DELETE.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/QueryName.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Http.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/GET.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Query.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/PATCH.java
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/PUT.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1199767211_1630583092.85/0/retrofit-2-8-1-sources-1-jar/retrofit2/Utils.java

1.1663 ansi-styles 4.3.0

1.1663.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1664 is-shared-array-buffer 1.0.1
1.1664.1 Available under license:
MIT License

Copyright (c) 2021 Inspect JS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1665 akka 2.5.29
1.1665.1 Available under license:
Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
This software includes the following components from the Apache Software Foundation, also released under the Apache Software License Version 2.0:

- commons-cli-1.1.jar
- jakarta-oro-2.0.8.jar
- log4j-1.2.15.jar

The following components are covered under different licenses:

- jline-0.9.94.jar (LICENSE-JLINE)
- One-Jar (LICENSE-ONEJAR)
- Java/Tcl v1.3.3 (LICENSE-TCLJAVA,-2,-3,-4)

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

---------------

Licenses for dependency projects can be found here:
[http://akka.io/docs/akka/snapshot/project/licenses.html]

---------------

akka-protobuf contains the sources of Google protobuf 2.5.0 runtime support, moved into the source package `akka.protobuf` so as to avoid version conflicts. For license information see COPYING.protobuf

The following license terms apply to the TJC compiler source and test files located in the src/tjc, src/tests/tjc, and tests/tjc directories.

2005 Advanced Micro Devices, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that existing copyright notices are retained in all copies, this notice is included verbatim in any distributions, and the terms and conditions hererin are met.

Use of the this software manifests acceptance of the terms of this license by performance.

The name of Advanced Micro Devices, Inc. may not be used to endorse or promote products derived from this software without specific prior
written permission.

THIS SOFTWARE IS PROVIDED BY ADVANCED MICRO DEVICES, INC. "AS IS" AND ANY EXPRESS, IMPLIED OR STATUTORY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT, OR THOSE ARISING FROM CUSTOM OF TRADE OR COURSE OF USAGE ARE DISCLAIMED.

IN NO EVENT SHALL ADVANCED MICRO DEVICES, INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED, AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE ITS DOCUMENTATION OR ANY DERIVATIVES THEREOF, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. BY USING THIS SOFTWARE WITHOUT CHARGE, YOU ACCEPT THIS ALLOCATION OF RISK. THIS DISCLAIMER OF LIABILITY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. ADVANCED MICRO DEVICES, INC. HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS OF THIS SOFTWARE.

In the redistribution and use of this software, each party shall at all times comply with all applicable governmental laws, statutes, ordinances, rules, regulations, orders, and other requirements, including without limitation such governmental requirements applicable to environmental protection, health, safety, wages, hours, equal employment opportunity, nondiscrimination, working conditions, import or export control, and transportation. Without limiting the foregoing, each party shall adhere to the U.S. Export Administration Regulations (EAR), currently found at 15 C.F.R. Sections 730 through 744, and, unless properly authorized by the U.S. Government, shall not (1) export, re-export or release restricted technology, software, or source code to a national of a country in Country Groups D:1 or E:1, or (2) export to Country Groups D:1 or E:1 the direct product of such technology or software, if such foreign produced direct product is subject to national security controls as identified on the Commerce Control List (currently found in Supplement 1 to Section 774 of EAR). These export requirements shall survive any expiration or termination of this agreement.

# Licenses

## Akka License

```
This software is licensed under the Apache 2 license, quoted below.


Licensed under the Apache License, Version 2.0 (the "License"); you may not
use this file except in compliance with the License. You may obtain a copy of
the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations under
the License.

---

## Akka Committer License Agreement

All committers have signed this [CLA](http://www.lightbend.com/contribute/current-cla).
It can be [signed online](http://www.lightbend.com/contribute/cla).

### Licenses for Dependency Libraries

Each dependency and its license can be seen in the project build file (the comment on the side of each dependency):
@extref[AkkaBuild.scala](github:project/AkkaBuild.scala#L1054)

Portions of Jal and Tcl Blend are
Copyright (c) 1997-1999 The Regents of the University of California.
All rights reserved.

Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the above
copyright notice and the following two paragraphs appear in all copies
of this software.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY
FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF
THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE
PROVIDED HEREBUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF
CALIFORNIA HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES,
ENHANCEMENTS, OR MODIFICATIONS.

The following license terms apply to the Itcl source and
test files located in the src/itcl and tests/itcl directories.

This software is copyrighted by Cadence Design Systems, Inc., and other
The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

Following is the original agreement for the Tcl/Tk software from Sun Microsystems.

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.
The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Commons CLI
Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
SUN MICROSYSTEMS, INC. THROUGH ITS SUN MICROSYSTEMS LABORATORIES
DIVISION ("SUN") WILL LICENSE THIS SOFTWARE AND THE ACCOMPANYING
DOCUMENTATION TO YOU (a "Licensee") ONLY ON YOUR ACCEPTANCE OF ALL
THE TERMS SET FORTH BELOW.

Sun grants Licensee a non-exclusive, royalty-free right to download,
install, compile, use, copy and distribute the Software, modify or
otherwise create derivative works from the Software (each, a
"Modification") and distribute any Modification in source code and/or
binary code form to its customers with a license agreement containing
these terms and noting that the Software has been modified. The
Software is copyrighted by Sun and other third parties and Licensee
shall retain and reproduce all copyright and other notices presently
on the Software. As between Sun and Licensee, Sun is the sole owner of
all rights in and to the Software other than the limited rights
granted to Licensee herein; Licensee will own its Modifications,
expressly subject to Sun's continuing ownership of the
Software. Licensee will, at its expense, defend and indemnify Sun and
its licensors from and against any third party claims, including costs
and reasonable attorneys' fees, and be wholly responsible for any
liabilities arising out of or related to Licensee's development, use
or distribution of the Software or Modifications. Any distribution of
the Software and Modifications must comply with all applicable United
States export control laws.
THE SOFTWARE IS BEING PROVIDED TO LICENSEE "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED. IN NO EVENT WILL SUN BE LIABLE HEREUNDER FOR ANY DIRECT DAMAGES OR ANY INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND.

The jmxsh library has been obtained from https://code.google.com/p/jmxsh/ under the Apache License 2.0. Copyright 2008, Google Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

/ * ====================================================================

* The Apache Software License, Version 1.1

*
* Copyright (c) 2000-2002 The Apache Software Foundation. All rights
* reserved.
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
* 3. The end-user documentation included with the redistribution,
*    if any, must include the following acknowledgment:
*    "This product includes software developed by the
*    Apache Software Foundation (http://www.apache.org/)."
*    Alternately, this acknowledgment may appear in the software itself,
*    if and wherever such third-party acknowledgments normally appear.
* 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
*    must not be used to endorse or promote products derived from this
*    software without prior written permission. For written
*    permission, please contact apache@apache.org.
* 5. Products derived from this software may not be called "Apache"
*    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
*    name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*/
/*
* One-JAR (http://www.simontuffs.com/one-jar). Copyright (c) 2004-2007, 
P. Simon Tuffs (simon@simontuffs.com). All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
* 
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
* 
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
* 
* Neither the name of P. Simon Tuffs, nor the names of any contributors,
* nor the name One-JAR may be used to endorse or promote products derived
* from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
*
* Including this file inside the built One-JAR file conforms with these terms.
*/

**1.1666 mime-db 1.47.0 1.47.0**

**1.1666.1 Available under license :**

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1667 immer 9.0.6 9.0.6

1.1667.1 Available under license:

MIT License

Copyright (c) 2017 Michel Weststrate

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1668 jul-to-slf4j 1.7.26
1.1669 jsreport-sample-template 2.6.1 2.6.1
1.1669.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1670 debug 3.1.0 3.1.0

1.1670.1 Available under license :
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1671 ncp 1.0.1 1.0.1
1.1671.1 Available under license :
# MIT License

###Copyright (C) 2011 by Charlie McConnell

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1672 commons-pool 2.6.0
1.1672.1 Available under license :
Apache Commons Pool
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

    Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS POOL DERIVATIVE WORKS:

The LinkedBlockingDeque implementation is based on an implementation written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/licenses/publicdomain

1.1673 source-map-support 0.5.21

1.1673.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Evan Wallace

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1674 poi-ooxml 5.0.0

1.1674.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

APACHE POI SUBCOMPONENTS:

Apache POI includes subcomponents with separate copyright notices and license terms. Your use of these subcomponents is subject to the terms and conditions of the following licenses:

Office Open XML schemas (poi-ooxml-full-*.jar)

The Office Open XML schema definitions used by Apache POI are a part of the Office Open XML ECMA Specification (ECMA-376, [1]). As defined in section 9.4 of the ECMA bylaws [2], this specification is available to all interested parties without restriction:

9.4 All documents when approved shall be made available to all interested parties without restriction.

Furthermore, both Microsoft and Adobe have granted patent licenses to this work [3,4,5].


Bouncy Castle library (bcprov-*.jar, bcpkg-*.jar, bcpkix-*.jar)

Copyright (c) 2000 - 2021 The Legion of the Bouncy Castle Inc.
(https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CurvesAPI / Curve API

BSD License

Copyright (c) 2000-2015 www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

SLF4J library (slf4j-api-*.jar)

Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

inbot-utils (https://github.com/Inbot/inbot-utils)

The MIT License (MIT)

Copyright (c) 2015 Inbot

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache POI

Copyright 2003-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This product contains parts that were originally based on software from BEA.
Copyright (c) 2000-2003, BEA Systems, <http://www.bea.com/> (dead link), which was acquired by Oracle Corporation in 2008.
This product contains W3C XML Schema documents. Copyright 2001-2003 (c) World Wide Web Consortium (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University)

This product contains the chunks_parse_cmds.tbl file from the vsdump program. Copyright (C) 2006-2007 Valek Filippov (frob@df.ru)


1.1675 ieee754 1.1.13 1.1.13

1.1675.1 Available under license:

Copyright 2008 Fair Oaks Labs, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1676 groovy-ant 3.0.8
1.1676.1 Available under license:

No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing,
software distributed under the License is distributed on an
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License for the
specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/groovy/ant/antlib.xml
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/antlib.xml
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/package.html

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/CompileTaskSupport.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/AntProjectPropertiesDelegate.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/RootLoaderRef.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/groovy/ant/FileNameFinder.groovy
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/UberCompileTask.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/groovy/ant/AntBuilder.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/FileSystemCompilerFacade.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/VerifyClass.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/Scanner.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/Groovy.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/Groovyc.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/GroovycTask.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/GenerateStubsTask.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/groovy/util/AntBuilder.java
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/groovy/util/FileNameFinder.groovy
* /opt/cola/permits/1261641276_1643305445.94/0/groovy-ant-3-0-8-sources-jar/org/codehaus/groovy/ant/Groovydoc.java

1.1677 compression 1.7.4 1.7.4

1.1677.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1678 frameguard 2.0.0 2.0.0

1.1678.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1679 json-simple 1.1.1
1.1680 setprototypeof 1.1.1 1.1.1

1.1680.1 Available under license:

Copyright (c) 2015, Wes Todd

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1681 lowercase-keys 2.0.0

1.1681.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1682 libtiff 4.0.6 1ubuntu0.7
1.1682.1 Available under license:
Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1683 jetty-client 9.3.27.v20190418

1.1684 multer 1.4.1 1.4.1
1.1684.1 Available under license:
Copyright (c) 2014 Hage Yaapa <http://www.hacksparrow.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT
OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1685 drill-format-xml 1.19.0

1.1685.1 Available under license:

Drill : Contrib : Format : XML
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1686 readdirp 3.6.0

1.1686.1 Available under license:

MIT License

Copyright (c) 2012-2019 Thorsten Lorenz, Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1687 psl 1.8.0 1.8.0

1.1687.1 Available under license:
The MIT License (MIT)

Copyright (c) 2017 Lupo Montero lupomontero@gmail.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1688 libxcursor 1.1.14-1ubuntu0.16.04.2

1.1688.1 Available under license:
Copyright 2002 Keith Packard

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Keith Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Keith Packard makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

KEITH PACKARD DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL KEITH PACKARD BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
1.1689 supports-color 2.0.0 2.0.0
1.1689.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1690 seamless-immutable 7.1.3 7.1.3
1.1690.1 Available under license :
Copyright (c) 2016, Richard Feldman
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of seamless-immutable nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1691 shimmer 1.0.0 1.0.0

1.1691.1 Available under license:

Copyright 2013, the Dart project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1692 safe-buffer 5.1.2 5.1.2
1.1692.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1693 circular-json 0.3.3
1.1693.1 Available under license:

*/

Copyright (C) 2013-2017 by Andrea Giammarchi - @WebReflection

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1694 batik-ttf2svg 1.13

1.1694.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software's
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1695 ret 0.2.2 0.2.2
1.1695.1 Available under license :

MIT License

Copyright (C) 2011 by fent

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1696 jsreport-browser-client-dist 1.3.0 1.3.0

1.1696.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1697 @babel/plugin-transform-template-literals 7.16.0

1.1697.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1698 argparse 0.7.0
1.1698.1 Available under license:

No license file was found, but licenses were detected in source scan.

# Redistribution and use in source and binary forms, with or without # modification, are permitted under the terms of the 3-clause BSD

Found in path(s):
* /opt/cola/permits/1000038828_1646171452.21/0/argparse-0-7-0-zip/argparse-0.7.0/setup.py
* /opt/cola/permits/1000038828_1646171452.21/0/argparse-0-7-0-zip/argparse-0.7.0/argparse.py
* /opt/cola/permits/1000038828_1646171452.21/0/argparse-0-7-0-zip/argparse-0.7.0/test_argparse.py

No license file was found, but licenses were detected in source scan.

Metadata-Version: 1.0
Name: argparse
Version: 0.7.0
Summary: An optparse-inspired command-line parsing library
Home-page: http://argparse.python-hosting.com/
Author: Steven Bethard
Author-email: steven.bethard@gmail.com
License: BSD
Description: Argparse takes the best of the optparse command-line parsing module and brings it new life. Argparse adds positional as well as optional arguments, the ability to create parsers for sub-commands, more informative help and usage messages, and much more. At the same time, it retains the ease and flexibility of use that made optparse so popular.
1.1699 extract-zip 1.7.0
1.1699.1 Available under license:

Copyright (c) 2014 Max Ogden and other contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1700 braces 3.0.2
1.1700.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1701 repeat-string 1.6.1 1.6.1

1.1701.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1702 annotations 3.0.1
1.1703 arr-diff 4.0.0

1.1703.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1704 reactapp 1.0.0 1.0.0

1.1704.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.

1.1705 babel-plugin-syntax-trailing-function-commas 6.22.0

1.1705.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The <Software identified by reference to the Ecma Standard* ("Software")> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1706 @babel/plugin-syntax-json-strings
7.8.3
1.1706.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1707.1 Available under license:

This package was debianized by Filip Van Raemdonck (mechanix@debian.org) on Thu, 4 Jul 2002 20:59:23 +0200.

It was downloaded from http://freedesktop.org/wiki/Software/shared-mime-info.

Copyright:
Copyright (C) 2002, 2003 Thomas Leonard.

License:

This package is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this package; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL'.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.1708 jline 3.12.1
1.1708.1 Available under license :

Copyright (c) 2002-2018, the original author or authors.
All rights reserved.

https://opensource.org/licenses/BSD-3-Clause

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following
conditions are met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with
the distribution.

Neither the name of JLine nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

1.1709 commons-compress 1.20
1.1709.1 Available under license:

Apache Commons Compress
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

---

The files in the package org.apachecommons.compress.archivers.sevenz
were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/),
which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)

---

The test file lbzip2_32767.bz2 has been copied from libbzip2's source
repository:
This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2019 Julian R Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1710 raml-parser 0.8.18 0.8.18

1.1710.1 Available under license:

    // Copyright Joyent, Inc. and other Node contributors.
    //
    // Permission is hereby granted, free of charge, to any person obtaining a
    // copy of this software and associated documentation files (the
    // "Software"), to deal in the Software without restriction, including
    // without limitation the rights to use, copy, modify, merge, publish,
    // distribute, sublicense, and/or sell copies of the Software, and to permit
    // persons to whom the Software is furnished to do so, subject to the
    // following conditions:
    //
    // The above copyright notice and this permission notice shall be included
    // in all copies or substantial portions of the Software.
    //
    // THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
    // OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
    // MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
    // NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
    // DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
    // OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
    // USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1711 is-binary-path 2.1.0 2.1.0

1.1711.1 Available under license:

    MIT License

    Copyright (c) 2019 Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com), Paul Miller
    (https://paulmillr.com)

    Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

    The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

    THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1712 jetty-xml 9.3.27.v20190418

1.1713 netty-transport-native-epoll

4.1.42.Final

1.1713.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */

/**
 * Set the {@code TCP_QUICKACK} option on the socket. See <a href="http://linux.die.net/man/7/tcp">TCP_QUICKACK</a> for more details.
 */

Found in path(s):
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollSocketChannelConfig.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 */
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  * /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-
    jar/io/netty/channel/epoll/Native.java
  * /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-
    jar/netty_epoll_native.c
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  * /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-
    jar/io/netty/channel/epoll/EpollDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollMode.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollRecvByteAllocatorStreamingHandle.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollChannelConfig.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */

Found in path(s):
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollChannelOption.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/package-info.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollSocketChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollServerSocketChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollEventLoopGroup.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollTcpInfo.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/AbstractEpollChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollEventLoop.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/NativeDatagramPacketArray.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/EpollDatagramChannel.java
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/Epoll.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty_epoll_linuxsocket.c
* /opt/cola/permits/1144318139_1615901439.63/0/netty-transport-native-epoll-4-1-42-final-sources-3-jar/io/netty/channel/epoll/NativeStaticallyReferencedJniMethods.java
1.1714 colors.js 1.0.3 1.0.3

1.1714.1 Available under license:

Original Library
- Copyright (c) Marak Squires

Additional Functionality
- Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1715 espree 3.5.4

1.1715.1 Available under license:

Espree
Copyright JS Foundation and other contributors, https://js.foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1716 body-parser 1.18.2 1.18.2
1.1716.1 Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1717 es-to-primitive 1.2.1
1.1717.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1718 avro-ipc-jetty 1.9.1

1.1718.1 Available under license:

Apache Avro IPC Jetty
Copyright 2009-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1719 jffi 1.0.1

1.1719.1 Available under license:

libffi - Copyright (c) 1996-2009 Anthony Green, Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
``Software''), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This code is free software: you can redistribute it and/or modify it under
the terms of the GNU Lesser General Public License version 3 only, as
published by the Free Software Foundation.

This code is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License
version 3 for more details.

You should have received a copy of the GNU Lesser General Public License
version 3 along with this work. If not, see <http://www.gnu.org/licenses/>.

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License” refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.
The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a
covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.
5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by
When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the
additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.
Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a
party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.1720 @babel/helper-function-name 7.10.4
1.1720.1 Available under license:
   MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1721 prop-types 15.7.2 15.7.2
1.1721.1 Available under license:
MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1722 metrics-servlets 4.0.2
1.1723 kerby-asn1 1.0.0

1.1723.1 Available under license:

Kerby ASN1 Project
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses...
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitaition, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1724 hive-spark-client 3.1.2
1.1724.1 Available under license :

Hive Spark Remote Client
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1725.1 Available under license:

MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1726 pump 3.0.0 3.0.0

1.1726.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1727 has-ansi 2.0.0
1.1727.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1728 aws-java-sdk-core 1.11.837
1.1728.1 Available under license :
Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are
under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of
fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source
code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
   2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

1.1729 babel-plugin-syntax-async-generators

6.13.0

1.1729.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1730 netty-transport-native-unix-common
4.1.42.Final
1.1730.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-
  jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-
  jar/io/netty/channel/unix/Limits.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-
  jar/netty_unix_util.h
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-
  jar/netty_unix_limits.h
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-jar/io/netty/channel/unix/UnixChannelOption.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4-jar/io/netty/channel/unix/IovArray.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4.jar/io/netty/channel/unix/UnixChannelUtil.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4-1-42-final-sources-4.jar/netty_unix_jni.h
No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_buffer.h
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_buffer.c

Found in path(s):
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_errors.h
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_errors.c
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_filedescriptor.c
* /opt/cola/permits/1144318138_1615901438.19/0/netty-transport-native-unix-common-4.1-42-final-sources-4.jar/netty_unix_filedescriptor.h
1.1731 jetty-servlet 9.4.34.v20201102

1.1731.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS
A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and  
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;  
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;  
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and  
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and  
b) a copy of this Agreement must be included with each copy of the Program.  
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and...
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
limitations under the License.

==============================================================================

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

==============================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

Open Source Used In DNAC Platform Arica 7458
If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1732 lazystream 1.0.0 1.0.0

1.1732.1 Available under license:

Copyright (c) 2013 J. Pommerening, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
1.1733 @babel/types 7.11.5

1.1733.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1734 jsreport-text 2.0.0 2.0.0

1.1734.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time...
a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1735 jersey-server 2.5.1

1.1735.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or
portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION.
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as “Participant”) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. " 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48
C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU
General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of
      any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
      Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
      License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started
      running for such interactive use in the most ordinary way, to print or display an announcement including an
      appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
      that users may redistribute the program under these conditions, and telling the user how to view a copy of this
      License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
      work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
      the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
      than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
      source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
      interchange; or,
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to
achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to
most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later
version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even
the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO
WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain
conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License.
Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright
disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at
compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program
is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If
this is what you want to do, use the GNU Library General Public License instead of this License.
/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved.

The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can obtain a copy of the License at http://glassfish.java.net/public/CDDL+GPL_1_1.html or packager/legal/LICENSE.txt. See the License for the specific language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields enclosed by brackets [] replaced by your own identifying information:
"Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or only the GPL Version 2, indicate your decision by adding "[Contributor] elects to include this software in this distribution under the [CDDL or GPL Version 2] license." If you don't indicate a single choice of license, a recipient has the option to distribute your version of this file under either the CDDL, the GPL Version 2 or to extend the choice of license to its licensees as provided above. However, if you add GPL Version 2 code and therefore, elected the GPL Version 2 license, then the option applies only if the new code is made subject to such option by the copyright holder.

1.1736 javax-inject 2.2.0-b21
1.1736.1 Available under license:
No license file was found, but licenses were detected in source scan.

2011 Oracle and/or its affiliates. All rights reserved.

The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development
and Distribution License ("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can obtain a copy of the License at https://glassfish.dev.java.net/public/CDDL+GPL_1_1.html or packager/legal/LICENSE.txt. See the License for the specific language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields enclosed by brackets [] replaced by your own identifying information:
"Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or only the GPL Version 2, indicate your decision by adding "[Contributor] elects to include this software in this distribution under the [CDDL or GPL Version 2] license." If you don't indicate a single choice of license, a recipient has the option to distribute your version of this file under either the CDDL, the GPL Version 2 or to extend the choice of license to its licensees as provided above. However, if you add GPL Version 2 code and therefore, elected the GPL Version 2 license, then the option applies only if the new code is made subject to such option by the copyright holder.

Found in path(s):
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The JSR-330 Expert Group
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/package-info.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Qualifier.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Singleton.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Provider.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Named.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Scope.java
* /opt/cola/permits/1136447767_1613767069.04/0/javax-inject-2-2-0-b21-sources-2-jar/javax/inject/Inject.java

1.1737 istanbuljs-load-nyc-config 1.1.0

1.1737.1 Available under license:

ISC License

Copyright (c) 2019, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.


1.1738.1 Available under license:

MIT License

Copyright (c) Facebook, Inc. and its affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1739 metrics 4.1.0

1.1739.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 MongoDB Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1740 ansi-escapes 1.4.0

1.1740.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1741 fbemitter 2.1.1 2.1.1
1.1741.1 Available under license :
   BSD License
For emitter software

Copyright (c) 2013-present, Facebook, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1742 whatwg-fetch 2.0.4 2.0.4
1.1742.1 Available under license :
   Copyright (c) 2014-2016 GitHub, Inc.
   Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1743 stdio 0.2.2

1.1743.1 Available under license :
(The MIT License)

Copyright (c) 2013- Sergio Garca <sgmonda@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1744 encoding 0.1.12 0.1.12
1.1744.1 Available under license:
Copyright (c) 2012-2014 Andris Reinman

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1745 jersey-client 2.25.1
1.1745.1 Available under license:
CDDL+GPL License
http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.1746 fast-safe-stringify 2.0.7 2.0.7
1.1746.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 David Mark Clements
Copyright (c) 2017 David Mark Clements & Matteo Collina
Copyright (c) 2018 David Mark Clements, Matteo Collina & Ruben Bridgewater

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1747 excel-streaming-reader 3.0.4
1.1747.1 Available under license:
No license file was found, but licenses were detected in source scan.

/* ====================================================================
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
==================================================================== */

Found in path(s):
* /opt/cola/permits/1258286463_1643052515.3/0/excel-streaming-reader-3-0-4-sources-jar/com/github/pjfanning/xlsx/impl/ooxml/XSSFReader.java

1.1748 elasticsearch-rest-client 7.0.1
1.1748.1 Available under license:
Elasticsearch
Copyright 2009-2018 Elasticsearch

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This product includes software developed by
Joda.org (http://www.joda.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1749 ini 1.3.5 1.3.5
1.1749.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1750 swagger-converter 1.5.1 1.5.1
1.1750.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Apigee Corporation

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1751 pause 0.0.1 0.0.1
1.1751.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1752 ajv 6.10.2 6.10.2
1.1752.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1753 mongo-java-driver 3.12.4

1.1753.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2012 The Netty Project
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/connection/netty/ReadTimeoutHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz and Tim Peierls
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 *
 * Any republication or derived work distributed in source code form
 * must include this copyright and license notice.
 */
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

---
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
*/
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/CreateIndexOptions.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/org/bson/codecs/pojo/PropertyMetadata.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/session/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/org/bson/codecs/pojo/PropertyAccessor.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/org/bson/json/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/changestream/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/vault/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/changestream/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/vault/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/changestream/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/client/model/vault/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
Open Source Used In DNAC Platform Arica 7500
jar/com/mongodb/GroupCommand.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/event/ConnectionPoolListenerAdapter.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/CommandReadOperation.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/MongoNamespace.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/MapReduceInlineResultsCursor.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/session/ServerSession.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/internal/connection/DnsSrvRecordMonitor.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/client/internal/MongoClientDelegate.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/org/bson/codecs/PatternCodec.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/CursorHelper.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/event/CommandEventMulticaster.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/MapReduceInlineResultsCursor.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/DropIndexOperation.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/org/bson/assertions/package-info.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/util/AbstractObjectSerializer.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/org/bson/json/JsonNullConverter.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/operation/DropDatabaseOperation.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/internal/connection/DefaultServerConnection.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/org/bson/codecs/BsonNullCodec.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/event/ServerHeartbeatSucceededEvent.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/connection/AsyncCompletionHandler.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/validator/NoOpFieldNameValidator.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/client/model/DBCollectionRemoveOptions.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
jar/com/mongodb/client/model/UpdateManyModel.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3.12.4-sources-
Open Source Used In DNAC Platform Arica 7528
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright (c) 2008-2014 Atlassian Pty Ltd
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/pojo/IdPropertyModelHolder.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/pojo/IdGenerator.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/pojo/ConventionObjectIdGeneratorsImpl.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/pojo/IdGenerators.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/com/mongodb/client/model/AggregationLevel.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/BsonCodecProvider.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/BsonCodec.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008-present MongoDB, Inc.
* Copyright 2018 Cezary Bartosiak
*Licensed under the Apache License, Version 2.0 (the "License");
*you may not use this file except in compliance with the License.
*You may obtain a copy of the License at
*http://www.apache.org/licenses/LICENSE-2.0
*
*Unless required by applicable law or agreed to in writing, software
*distributed under the License is distributed on an "AS IS" BASIS,
*WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*See the License for the specific language governing permissions and
*limitations under the License.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/jsr310/InstantCodec.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/jsr310/Jsr310CodecProvider.java
*/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/org/bson/codecs/jsr310/LocalDateTimeCodec.java
No license file was found, but licenses were detected in source scan.

/\n */
* Copyright 2008-present MongoDB, Inc.
 *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
 *
*   http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
 */
/\n*/
/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/
/**
 * Returns an array containing all of the elements in this deque, in
 * proper sequence (from first to last element).
 *
 * <p>The returned array will be "safe" in that no references to it are
 * maintained by this deque. (In other words, this method must allocate
 * a new array). The caller is thus free to modify the returned array.
 *
 * <p>This method acts as bridge between array-based and collection-based
 * APIs.
 *
 * @return an array containing all of the elements in this deque
 */

No license file was found, but licenses were detected in source scan.

/\n */
* Copyright 2008-present MongoDB, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* 
* Original Work: MIT License, Copyright (c) [2015-2018] all contributors
* https://github.com/marianobarrios/tls-channel
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/TlsChannelBuilder.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannelGroup.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/async/WouldBlockException.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannel.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/ExtendedAsynchronousByteChannel.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/BufferAllocator.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/impl/BufferHolder.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferSet.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferUtil.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/impl/TlsChannelFlowControlException.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferUtil.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/TrackingAllocator.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/TlsChannelImpl.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/TlsChannelCallbackException.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/ClientTlsChannel.java
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2017 Tom Bentley
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):

jar/com/mongodb/internal/connection/tlschannel/NeedsReadException.java
/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/com/mongodb/internal/connection/tlschannel/NeedsWriteException.java
/opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-
jar/com/mongodb/internal/connection/tlschannel/NeedsTaskException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2010 The Guava Authors
 * Copyright 2011 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original Work: MIT License, Copyright (c) [2015-2018] all contributors
 * https://github.com/marianobarrios/tls-channel
 */
/**
 * Writes a sequence of BytesProduced to this channel from a subsequence of the
 * given buffers.
 *
 * <p>
 * See {@link GatheringByteChannel#write(ByteBuffer[], int, int)} for more
 * details of the meaning of this signature.
 *
 * <p>
 * This method behaves slightly different than the interface specification,
 * with respect to non-blocking responses, see {@link #write(ByteBuffer)}
 * for more details.
 *
 * @param srcs   The buffers from which BytesProduced are to be retrieved
 * @param offset The offset within the buffer array of the first buffer from
 *               which BytesProduced are to be retrieved; must be non-negative and no
 *               larger than <tt>srcs.length</tt>
 * @param length The maximum number of buffers to be accessed; must be
 *               non-negative and no larger than <tt>srcs.length</tt>
 * @return The number of BytesProduced written, contrary to the behavior specified
 * in {@link ByteChannel}, this method never returns 0, but throws
 * [ @link WouldBlockException ]
 * @throws IndexOutOfBoundsException If the preconditions on the <tt>offset</tt> and

* @throws WouldBlockException if the channel is in non-blocking mode and the IO operation cannot be completed immediately
* @throws NeedsTaskException if the channel is not configured to run tasks automatically and a task needs to be executed to complete the operation
* @throws SSLException if the {@link SSLEngine} throws a SSLException
* @throws IOException if the underlying channel throws an IOException
*/

/**
* Reads a sequence of bytesProduced from this channel into a subsequence of the given buffers.
*
* See {@link ScatteringByteChannel#read(ByteBuffer[], int, int)} for more details of the meaning of this signature.
*
* This method behaves slightly different than the interface specification, with respect to non-blocking responses, see {@link #read(ByteBuffer)} for more details.
*
* @param dsts   The buffers into which bytesProduced are to be transferred
* @param offset The offset within the buffer array of the first buffer into which bytesProduced are to be transferred; must be non-negative and no larger than <tt>dsts.length</tt>
* @param length The maximum number of buffers to be accessed; must be non-negative and no larger than <tt>dsts.length</tt>&nbsp;-&nbsp;<tt>offset</tt>
* @return The number of bytesProduced read, or &lt;tt&gt;-1&lt;/tt&gt; if the channel has reached end-of-stream; contrary to the behavior specified in [ @link ByteChannel], this method never returns 0, but throws [ @link WouldBlockException]
* @throws IndexOutOfBoundsException If the preconditions on the <tt>offset</tt> and <tt>length</tt> parameters do not hold
* @throws WouldBlockException if the channel is in non-blocking mode and the IO operation cannot be completed immediately
* @throws NeedsTaskException if the channel is not configured to run tasks automatically and a task needs to be executed to complete the operation
* @throws SSLException if the {@link SSLEngine} throws a SSLException
* @throws IOException if the underlying channel throws an IOException
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1065171190_1593486559.87/0/mongo-java-driver-3-12-4-sources-jar/com/mongodb/internal/connection/tlschannel/TlsChannel.java
1.1754 unzipper 0.9.15 0.9.15

1.1754.1 Available under license:

Copyright (c) 2012 - 2013 Near Infinity Corporation

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

Commits in this fork are (c) Ziggy Jonsson (ziggy.jonsson.nyc@gmail.com)
and fall under same licence structure as the original repo (MIT)

1.1755 jsreport-authentication 2.6.1 2.6.1

1.1755.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1756 emoji-regex 7.0.3
1.1756.1 Available under license :
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1757 reflectasm 1.10.1
1.1757.1 Available under license :
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2008, Nathan Sweet
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of Esoteric Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):
* /opt/cola/permits/1145449535_1616181967.19/0/reflectasm-1-10-1-sources-3-jar/com/esotericsoftware/reflectasm/ConstructorAccess.java
* /opt/cola/permits/1145449535_1616181967.19/0/reflectasm-1-10-1-sources-3-jar/com/esotericsoftware/reflectasm/PublicConstructorAccess.java
* /opt/cola/permits/1145449535_1616181967.19/0/reflectasm-1-10-1-sources-3-jar/com/esotericsoftware/reflectasm/MethodAccess.java
* /opt/cola/permits/1145449535_1616181967.19/0/reflectasm-1-10-1-sources-3-jar/com/esotericsoftware/reflectasm/FieldAccess.java
* /opt/cola/permits/1145449535_1616181967.19/0/reflectasm-1-10-1-sources-3-jar/com/esotericsoftware/reflectasm/AccessClassLoader.java

---

1.1758 babel-plugin-transform-es2015-destructuring 6.23.0

1.1758.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1759.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1760 saslprep 1.0.3 1.0.3
1.1760.1 Available under license:
Copyright (c) 2014 Dmitry Tsvettsikh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1761 atk 2.18.1-2ubuntu1
1.1761.1 Available under license:

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of
this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.
Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The
former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.
In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or
link a “work that uses the Library” with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable “work that
uses the Library”, as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.1762 mesa 18.0.5-0ubuntu0~16.04.1
1.1762.1 Available under license:
Legal characters in GLSL are:

* Identifier characters:
  * Letters a-z
  * Letters A-Z
  * Underscore
  * Numbers 0-9

* Punctuation:
* Period, plus, dash, slash, asterisk, percent, angled brackets,
* square brackets, parentheses, braces, caret, vertical bar,
* ampersand, tilde, equals, exclamation point, colon, semicolon,
* comma, and question mark
* Special:
*   * Number sign (as used in preprocessor)
*   * Backslash just before newline as line continuation
*   * White space:
*     * Space, horizontal tab, vertical tab, form feed, carriage-return,
*       * and line-feed.
*   * [GLSL Language Specification 4.30.6, section 3.1]
* In this file, we test each of these in turn as follows:
*   * Identifier characters: All pass through unchanged
*   * Punctuation: All pass through unchanged
*   * Special: Empty directive replaced with blank line
*     * Line continuation merges two lines, then a blank line
*   * Whitespace: 4 horizontal space characters each replaced with space
*     * 2 newline characters each replaced with a newline
* */
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKMLNOPQRSTUVWXYZ
_0123456789
.
+
-
/
*
%
<
>
[
]
(
)
{
}
^
# Copyright IBM Corporation 2004
# All Rights Reserved.
#
# Permission is hereby granted, free of charge, to any person obtaining a
# copy of this software and associated documentation files (the "Software"),
# to deal in the Software without restriction, including without limitation
# on the rights to use, copy, modify, merge, publish, distribute, sub
# license, and/or sell copies of the Software, and to permit persons to whom
# the Software is furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice (including the next
# paragraph) shall be included in all copies or substantial portions of the
# Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
# IBM AND/OR ITS SUPPLIERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
# IN THE SOFTWARE.
#
# Authors:
#  Ian Romanick <idr@us.ibm.com>

bsd_license_template = """%s
All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sub license, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL %s, AND/OR THEIR SUPPLIERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE."
Please do not refer to the library as <em>MesaGL</em> (for legal reasons). It's just <em>Mesa</em> or <em>The Mesa 3-D graphics library</em>. <br>


<h1>License / Copyright Information</h1>

The Mesa distribution consists of several components. Different copyrights and licenses apply to different components. For example, the GLX client code uses the SGI Free Software License B, and some of the Mesa device drivers are copyrighted by their authors. See below for a list of Mesa's main components and the license for each.

The core Mesa library is licensed according to the terms of the MIT license. This allows integration with the XFree86, Xorg and DRI projects.

The default Mesa license is as follows:

Copyright (C) 1999-2007  Brian Paul  All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
Attention, Contributors

When contributing to the Mesa project you must agree to the licensing terms of the component to which you're contributing. The following section lists the primary components of the Mesa distribution and their respective licenses.

Mesa Component Licenses

<table>
<thead>
<tr>
<th>Component</th>
<th>Location</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Mesa code</td>
<td>src/mesa/</td>
<td>MIT</td>
</tr>
<tr>
<td>Device drivers</td>
<td>src/mesa/drivers/*</td>
<td>MIT, generally</td>
</tr>
<tr>
<td>Gallium code</td>
<td>src/gallium/</td>
<td>MIT</td>
</tr>
<tr>
<td>Ext headers</td>
<td>include/GL/glext.h</td>
<td>Khronos</td>
</tr>
<tr>
<td></td>
<td>include/GL/glxext.h</td>
<td></td>
</tr>
<tr>
<td>GLX client code</td>
<td>src/glx/</td>
<td>SGI Free Software License B</td>
</tr>
<tr>
<td>C11 thread</td>
<td>include/c11/threads*.h</td>
<td>Boost (permissive)</td>
</tr>
</tbody>
</table>

In general, consult the source files for license terms.
1.1763.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1764 scala 2.11.12

1.1764.1 Available under license:

Scala includes the Sizzle library:

Copyright (c) 2010 The Dojo Foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Scala is licensed under the [BSD 3-Clause License](http://opensource.org/licenses/BSD-3-Clause).

## Scala License

Copyright (c) 2002-2017 EPFL

Copyright (c) 2011-2017 Lightbend, Inc.

All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the EPFL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Other Licenses

This software includes projects with the following licenses, which are also included in the `licenses/` directory:

### [Apache License](http://www.apache.org/licenses/LICENSE-2.0.html)
This license is used by the following third-party libraries:

* jansi

### [BSD License](http://www.opensource.org/licenses/bsd-license.php)
This license is used by the following third-party libraries:

* jline

### [BSD 3-Clause License](http://opensource.org/licenses/BSD-3-Clause)
This license is used by the following third-party libraries:

* asm

### [MIT License](http://www.opensource.org/licenses/MIT)
This license is used by the following third-party libraries:
* jquery
* jquery-ui
* jquery-layout
* sizzle
* tools tooltip

### Public Domain
The following libraries are freely available in the public domain:

* forkjoin

Scala includes the JLine library:

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following
conditions are met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with
the distribution.

Neither the name of JLine nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2011 Paul Bakaus, http://jqueryui.com/

This software consists of voluntary contributions made by many
individuals (AUTHORS.txt, http://jqueryui.com/about) For exact
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Scala includes the ASM library.

Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

not-a-legal-formal-parameter-tuple.scala:2: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
   Either create a single parameter accepting the Tuple2,
   or consider a pattern matching anonymous function: `case (a, b) => ...
   val x: ((Int, Int) => Int) = (((a, b)) => a)
^  
not-a-legal-formal-parameter-tuple.scala:3: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
   Either create a single parameter accepting the Tuple2,
   or consider a pattern matching anonymous function: `case (param1, param2) => ...
   val y: ((Int, Int, Int) => Int) = (((a, !)) => a)
^  
not-a-legal-formal-parameter-tuple.scala:4: error: not a legal formal parameter.
Note: Tuples cannot be directly destructured in method or function parameters.
   Either create a single parameter accepting the Tuple3,
   or consider a pattern matching anonymous function: `case (param1, ..., param3) => ...
   val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)
^  
three errors found

Scala includes the Tools Tooltip library:

Copyright (c) 2009 Tero Piirainen
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(c) 2012-2014 GitHub

When using the GitHub logos, be sure to follow the GitHub logo guidelines (https://github.com/logos)

Font License: SIL OFL 1.1 (http://scripts.sil.org/OFL)
Applies to all font files

Code License: MIT (http://choosealicense.com/licenses/mit/)
Applies to all other files
Copyright (c) 2006, Ivan Sagalaev
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of highlight.js nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Scala includes the JLine library, which includes the Jansi library.
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Scala includes the jQuery library:

Copyright (c) 2010 John Resig

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(The MIT License)
Copyright (c) 2013 Greg Allen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1765 nopt 3.0.6 3.0.6

1.1765.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1766 ms 2.1.1 2.1.1
1.1766.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1767 bson 4.2.3
1.1767.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licenser" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1768 @babel/plugin-proposal-private-methods 7.16.0

1.1768.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1769 jetty-io 9.3.27.v20190418

1.1769.1 Available under license :

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and
ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular
Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself
or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are
separate modules of software distributed in conjunction with the Program under their own license agreement, and
(ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or
sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide,
royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and
object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide,
royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the
combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall
not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no
assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any
other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights
and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual
property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute
the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to
grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided
that:
a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively “Losses”) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor’s responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was
received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.1770 @babel/helper-split-export-declaration
7.16.0
1.1770.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1771 jackson-jaxrs-xml-provider 2.10.4

1.1771.1 Available under license :

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.1772 commons-dbcp 1.4

1.1772.1 Available under license :

Apache Commons DBCP
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1773 node-readfiles 0.2.0 0.2.0

1.1773.1 Available under license:

(The MIT License)

Copyright (c) 2016 Alejandro Gonzalez <guatedude2@hotmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1774 libxkbcommon 0.5.0-1ubuntu2.1

1.1774.1 Available under license:

The following is a list of all copyright notices and license statements which appear in the xkbcommon source tree.

If making new contributions, the first form (i.e. Daniel Stone, Ran Benita, etc) is vastly preferred.

All licenses are derivative of the MIT/X11 license, mostly identical other than no-endorsement clauses (e.g. paragraph 4 of The Open Group's license).

These statements are split into two sections: one for the code compiled and distributed as part of the libxkbcommon shared library and the code.
component of all tests (i.e. everything under src/ and xkbcommon/, plus the .c and .h files under test/), and another for the test data under test/data, which is distributed with the xkbcommon source tarball, but not installed to the system.

BEGINNING OF SOFTWARE COPYRIGHT/LICENSE STATEMENTS:

-------------------------------------------------------------------------------
Copyright 2009-2012 Daniel Stone
Copyright 2012 Ran Benita <ran234@gmail.com>
Copyright 2010, 2012 Intel Corporation
Copyright 2008, 2009 Dan Nicholson
Copyright 2010 Francisco Jerez <currojerez@riseup.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-------------------------------------------------------------------------------


Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or their institutions shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the authors.

-------------------------------------------------------------------------------


Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Silicon Graphics not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission. Silicon Graphics makes no representation about the suitability of this software for any purpose. It is provided "as is" without any express or implied warranty.

SILICON GRAPHICS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

-------------------------------------------------------------------------------

Copyright 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts.
Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Digital not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

DIGITAL DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL
DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR
ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.

------------------------------------------------------------------------

Copyright (C) 2011 Joseph Adams <joeyadams3.14159@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

------------------------------------------------------------------------
END OF SOFTWARE COPYRIGHT/LICENSE STATEMENTS

BEGINNING OF LICENSE STATEMENTS FOR UNDISTRIBUTED DATA FILES IN test/data, derived from xkeyboard-config:

-------------------------------------------------------------------------------
Copyright 1996 by Joseph Moss
Copyright (C) 2002-2007 Free Software Foundation, Inc.
Copyright (C) Dmitry Golubev <lastguru@mail.ru>, 2003-2004
Copyright (C) 2004, Gregory Mokhin <mokhin@bog.msu.ru>
Copyright (C) 2006 Erdal Ronah

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of the copyright holder(s) not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The copyright holder(s) makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THE COPYRIGHT HOLDER(S) DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THE COPYRIGHT HOLDER(S) BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

-------------------------------------------------------------------------------
Copyright 1992 by Oki Technosystems Laboratory, Inc.
Copyright 1992 by Fuji Xerox Co., Ltd.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Oki Technosystems Laboratory and Fuji Xerox not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.
Oki Technosystems Laboratory and Fuji Xerox make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1775 jedis 2.10.2-cisco

1.1775.1 Available under license:

Copyright (c) 2010 Jonathan Leibiusky

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1776 @babel/helper-remap-async-to-generator 7.16.4
1.1776.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1777 jsr305 3.0.1

1.1778 buffer-equal-constant-time 1.0.1 1.0.1

1.1778.1 Available under license:

Copyright (c) 2013, GoInstant Inc., a salesforce.com company
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of salesforce.com, nor GoInstant, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1779 babel-plugin-transform-reserved-words 7.16.0
1.1779.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1780 node-ensure 0.0.0 0.0.0
1.1780.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1781 yargs 13.2.4 13.2.4
1.1781.1 Available under license :

Copyright 2010 James Halliday (mail@substack.net)
Modified work Copyright 2014 Contributors (ben@npmjs.com)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1782 babel-plugin-transform-es2015-template-literals 6.22.0

1.1782.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The "Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1783 xml-apis 1.4.01

1.1783.1 Available under license:

xml-commons/java/external/LICENSE.dom-software.txt $Id: LICENSE.dom-software.txt 734314 2009-01-14 03:33:27Z mrglavas $

This license came from: http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.zip (COPYRIGHT.html)

W3C SOFTWARE NOTICE AND LICENSE
All Rights Reserved.

The DOM bindings are published under the W3C Software Copyright Notice and License. The software license requires "Notice of any changes or modifications to the W3C files, including the date changes were made." Consequently, modified versions of the DOM bindings must document that they do not conform to the W3C standard; in the case of the IDL definitions, the pragma prefix can no longer be 'w3c.org'; in the case of the Java language binding, the package names can no longer be in the 'org.w3c' package.

Note: The original version of the W3C Software Copyright Notice and License could be found at http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.

3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache xml-commons xml-apis ==
== distribution. ==
========================================================================

Apache XML Commons XML APIs
Copyright 1999-2009 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org

Note: The original version of the W3C Software Copyright Notice and License could be found at http://www.w3.org/Consortium/Legal/copyright-software-19980720

Permission to use, copy, and modify this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body
of any redistributed or derivative code: "Copyright 2002
Informatique et en Automatique</a>, <a href="http://www.keio.ac.jp/">Keio University</a>). All Rights Reserved.

http://www.w3.org/Consortium/Legal/"

Notice of any changes or modifications to the W3C files, including the
date changes were made. (We recommend you provide URIs to the location
from which the code is derived.)</li>
</ol>

<p>THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT
LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY
THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.</p>

<p>COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR
DOCUMENTATION.</p>

The name and trademarks of copyright holders may NOT be used in advertising
or publicity pertaining to the software without specific, written prior
permission. Title to copyright in this software and any associated
documentation will at all times remain with copyright holders.</p>

This license came from: http://www.w3.org/Consortium/Legal/copyright-documents-20021231

W3C DOCUMENT LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231

Public documents on the W3C site are provided by the copyright holders under
the following license. By using and/or copying this document, or the W3C
document from which this statement is linked, you (the licensee) agree that
you have read, understood, and will comply with the following terms and
conditions:

Permission to copy, and distribute the contents of this document, or the W3C
document from which this statement is linked, in any medium for any purpose
and without fee or royalty is hereby granted, provided that you include the
following on ALL copies of the document, or portions thereof, that you use:
or annotating specifications. Other questions about this notice can be directed
to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) $

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered
under http://saxproject.org/?selected=pd

This page is now out of date -- see the new SAX site at
http://www.saxproject.org/ for more up-to-date
releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for
XML), and release all of the SAX 2.0 source code, compiled code, and
documentation contained in this distribution into the Public Domain.
SAX comes with NO WARRANTY or guarantee of fitness for any
purpose.

David Megginson, david@megginson.com
2000-05-05

1.1784 jetty 9.3.27.v20190418

### 1.1784.1 Available under license:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and
ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:
a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was
received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.1785 babel-helper-bindify-decorators 6.24.1
1.1785.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1786.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Apache Commons Text
Copyright 2014-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

1.1787 paranamer 2.3
1.1787.1 Available under license :

* Copyright (c) 2007 Paul Hammant
* All rights reserved.
*   * Redistribution and use in source and binary forms, with or without
*     modification, are permitted provided that the following conditions
*     are met:
*     * 1. Redistributions of source code must retain the above copyright
*        notice, this list of conditions and the following disclaimer.
*     * 2. Redistributions in binary form must reproduce the above copyright
*        notice, this list of conditions and the following disclaimer in the
*        documentation and/or other materials provided with the distribution.
*     * 3. Neither the name of the copyright holders nor the names of its
*        contributors may be used to endorse or promote products derived from
*        this software without specific prior written permission.
*     *
*     * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
*     AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
*     IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
*     ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
*     LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
*     CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
*     SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
*     INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
*     CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
*     ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
*     THE POSSIBILITY OF SUCH DAMAGE.
*/
1.1788 eslint 3.19.0

1.1788.1 Available under license:

ESLint
Copyright JS Foundation and other contributors, https://js.foundation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1789 jackson-databind 2.12.3

1.1789.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.1790 commons-configuration2 2.1.1

1.1790.1 Available under license :
Apache Commons Configuration
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, 
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by 
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all 
other entities that control, are controlled by, or are under common 
control with that entity. For the purposes of this definition, 
"control" means (i) the power, direct or indirect, to cause the 
direction or management of such entity, whether by contract or 
otherwise, or (ii) ownership of fifty percent (50%) or more of the 
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity 
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, 
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1791 jaxb-api 2.2.11

1.1791.1 Available under license:

Found license 'General Public License 2.0' in "Copyright (c) 2004-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'Common Development and Distribution License (CDDL) 1.1’ in 'Copyright (c) 2003-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License’

Found license 'General Public License 2.0' in 'Copyright (c) 2004-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only
("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2006-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2003-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2005-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2007-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2004-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License.

Found license 'General Public License 2.0' in Copyright (c) 2003-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may
not use this file except in compliance with the License. You can use this file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License

1.1792 commons-io 2.6
1.1792.1 Available under license:
Apache Commons IO
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1793 de.huxhorn.lilith.logback.converter-classic 0.9.44

1.1793.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the Library.

/*
 * Lilith - a log event viewer.
 * Copyright (C) 2007-2011 Joern Huxhorn
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU Lesser General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/*
 * Copyright 2007-2011 Joern Huxhorn
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

1.1794 uglify-js 3.11.3 3.11.3

1.1794.1 Available under license:

UglifyJS is released under the BSD license:

Copyright 2012-2019 (c) Mihai Bazon <mihai.bazon@gmail.com>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1795 qs 6.5.2 6.5.2
1.1795.1 Available under license:

Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors

1.1796 jnr-posix 1.1.4

1.1796.1 Available under license:

jnr-posix is released under a tri CPL/GPL/LGPL license. You can use it, redistribute it and/or modify it under the terms of the license of your choice.

=== Common Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents ” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:
a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS),
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

=== GPL 2.0

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and “any
later version”, you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

== GNU LGPL 2.1

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistributions and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1797 woodstox-core 5.0.3

1.1797.1 Available under license:

No license file was found, but licenses were detected in source scan.

   <name>The Apache License, Version 2.0</name>
   <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1135953508_1613640887.59/0/woodstox-core-5-0-3-3-jar/META-INF/maven/com.fasterxml.woodstox/woodstox-core/pom.xml

No license file was found, but licenses were detected in source scan.

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.6))"
Specification-Title: Woodstox
Specification-Vendor: FasterXML
Specification-Version: 5.0.3
Tool: Bnd-2.3.0.201405100607
X-Compile-Source-JDK: 1.6
X-Compile-Target-JDK: 1.6

Found in path(s):
* /opt/cola/permits/1135953508_1613640887.59/0/woodstox-core-5-0-3-3-jar/META-INF/MANIFEST.MF

1.1798 is-path-inside 3.0.3
1.1798.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1799 kotlin 1.4.0

1.1799.1 Available under license:

No license file was found, but licenses were detected in source scan.
w3c.dom.poinerevents.kt","org.w3c/org.w3c.dom.svg.kt","org.w3c/org.w3c.files.kt","org.w3c/org.w3c.notification
s.kt","org.w3c/org.w3c.workers.kt","org.w3c/org.w3c.xhr.kt","annotations/Experimental.kt","annotations/Experime
tnalStdlibApi.kt","annotations/Inference.kt","annotations/Multiplatform.kt","annotations/OptIn.kt","collections/Abstr
actCollection.kt","collections/AbstractIterator.kt","collections/AbstractList.kt","collections/AbstractMap.kt","collections/AbstractSet.kt","collections/ArrayDeque.kt","collections/Arrays.kt","collections/IndexedValue.kt","collections /MapAccessors.kt","collections/MapWithDefault.kt","collections/Maplike.kt","collections/MapWithDefault.kt","collections/MutableCollections.kt","collections/ReversedVie
ws.kt","collections/SequenceBuilder.kt","collections/SlidingWindow.kt","collections/UnaryOperator.kt","contracts/
ContractBuilder.kt","coroutines/ContinuationInterceptor.kt","coroutines/CoroutineContext.kt","coroutines/Coroutin

insertContent();

Kotlin.isBooleanArray = function (a) {
    return Array.isArray(a) || a instanceof Int8Array && a.$type$ === "BooleanArray"
};

Kotlin.isByteArray = function (a) {
    return a instanceof Int8Array && a.$type$ !== "BooleanArray"
};

Kotlin.isShortArray = function (a) {
    return a instanceof Int16Array
};

Kotlin.isCharArray = function (a) {
    return a instanceof Uint16Array && a.$type$ === "CharArray"
};

Kotlin.isIntArray = function (a) {
    return a instanceof Int32Array
};

Kotlin.isFloatArray = function (a) {
    return a instanceof Float32Array
};

Kotlin.isDoubleArray = function (a) {
    return a instanceof Float64Array
};

Kotlin.isLongArray = function (a) {
    return Array.isArray(a) && a.$type$ === "LongArray"
};

Kotlin.arrayToString = function (a) {
    if (a === null) return "null"
    var toString = Kotlin.isCharArray(a) ? String.fromCharCode : Kotlin.toString;
    return "[" + Array.prototype.map.call(a, function(e) { return toString(e); }).join(", ") + "]";

    Kotlin.arrayDeepToString = function (arr) {
        return Kotlin.kotlin.collections.contentDeepToStringImpl(arr);
    }

    Kotlin.arrayEquals = function (a, b) {
        if (a === b) {
            return true;
        }
        if (a === null || b === null || !Kotlin.isArrayish(b) || a.length !== b.length) {
            return false;
        }
        for (var i = 0; i < a.length; i++) {
            if (!Kotlin.equals(a[i], b[i])) {
                return false;
            }
        }
        return true;
    }

    Kotlin.arrayDeepEquals = function (a, b) {
        if (a === b) {
            return true;
        }
        if (a === null && b === null) {
            return true;
        }
        if (a.length !== b.length) {
            return false;
        }
        for (var i = 0; i < a.length; i++) {
            if (!Kotlin.equals(a[i], b[i])) {
                return false;
            }
        }
        return true;
    }

    Kotlin.arrayHashCode = function (arr) {
        if (arr === null) return 0
        var result = 1;
        for (var i = 0, n = arr.length; i < n; i++) {
            result = (31 * result | 0) + Kotlin.hashCode(arr[i]) | 0;
        }
        return result;
    }

    Kotlin.arrayDeepHashCode = function (arr) {
        return Kotlin.kotlin.collections.contentDeepHashCodeImpl(arr);
    }

    Kotlin.isArray = function (a) {
        return Array.isArray(a) && !a.$type$;
    }

    Kotlin.isArrayish = function (a) {
        return Array.isArray(a) || ArrayBuffer.isView(a)
    }

    Kotlin.primitiveArraySort = function (array) {
        array.sort(Kotlin.doubleCompareTo)
    }

    Kotlin.getCallableRef = function(name, f) {
        f.callableName = name;
        return f;
    }

    Kotlin.getPropertyCallableRef = function(name, paramCount, getter, setter) {
        return this.propertyCallableRef(name, paramCount, getter, setter);
    }

    return this;
} });

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

Kotlin.arrayToString = function (a) {
    if (a === null) return "null"
    var toString = Kotlin.isCharArray(a) ? String.fromCharCode : Kotlin.toString;
    return "[" + Array.prototype.map.call(a, function(e) { return toString(e); }).join(", ") + "]";
}

Kotlin.arrayDeepToString = function (arr) {
    return Kotlin.kotlin.collections.contentDeepToStringImpl(arr);
}

Kotlin.arrayEquals = function (a, b) {
    if (a === b) {
        return true;
    }
    if (a === null || b === null || !Kotlin.isArrayish(b) || a.length !== b.length) {
        return false;
    }
    for (var i = 0; i < a.length; i++) {
        if (!Kotlin.equals(a[i], b[i])) {
            return false;
        }
    }
    return true;
}

Kotlin.arrayDeepEquals = function (a, b) {
    if (a === b) {
        return true;
    }
    if (a === null && b === null) {
        return true;
    }
    if (a.length !== b.length) {
        return false;
    }
    for (var i = 0; i < a.length; i++) {
        if (!Kotlin.equals(a[i], b[i])) {
            return false;
        }
    }
    return true;
}

Kotlin.arrayHashCode = function (arr) {
    if (arr === null) return 0
    var result = 1;
    for (var i = 0, n = arr.length; i < n; i++) {
        result = (31 * result | 0) + Kotlin.hashCode(arr[i]) | 0;
    }
    return result;
}

Kotlin.arrayDeepHashCode = function (arr) {
    return Kotlin.kotlin.collections.contentDeepHashCodeImpl(arr);
}

Kotlin.primitiveArraySort = function (array) {
    array.sort(Kotlin.doubleCompareTo)
} });

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

Kotlin.getCallableRef = function(name, f) {
    f.callableName = name;
    return f;
}

Kotlin.getPropertyCallableRef = function(name, paramCount, getter, setter) {
    return this.propertyCallableRef(name, paramCount, getter, setter);
}
getter.set = setter;
getter.callableName = name;
return getPropertyRefClass(getter, setter, propertyRefClassMetadataCache[paramCount]);
}

function getPropertyRefClass(obj, setter, cache) {
    if (cache.value === null) {
        cache.value = { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutuableProperty0 } };
    }
    return cache.value;
}

function getPropertyRefClass(obj, setter, cache) {
    if (cache.value === null) {
        cache.value = { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutuableProperty0 } };
    }
    return cache.value;
}

getter.set = setter;
getter.callableName = name;
return getPropertyRefClass(getter, setter, propertyRefClassMetadataCache[paramCount]);
}

function getPropertyRefClass(obj, setter, cache) {
    if (cache.value === null) {
        cache.value = { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutuableProperty0 } };
    }
    return cache.value;
}

getter.set = setter;
getter.callableName = name;
return getPropertyRefClass(getter, setter, propertyRefClassMetadataCache[paramCount]);
}

function getPropertyRefClass(obj, setter, cache) {
    if (cache.value === null) {
        cache.value = { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutuableProperty0 } };
    }
    return cache.value;
}

getter.set = setter;
getter.callableName = name;
return getPropertyRefClass(getter, setter, propertyRefClassMetadataCache[paramCount]);
}

function getPropertyRefClass(obj, setter, cache) {
    if (cache.value === null) {
        cache.value = { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutuableProperty0 } };
    }
    return cache.value;
}
the Apache 2.0 license that can be found in the license/LICENSE.txt file. You may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

// Copyright 2009 The Closure Library Authors. All Rights Reserved.
*/

// Copyright 2009 The Closure Library Authors. All Rights Reserved.

/**
 * Constructs a 64-bit two's-complement integer, given its low and high 32-bit values as *signed* integers. See the from* functions below for more convenient ways of constructing Longs.
 * The internal representation of a long is the two given signed, 32-bit values.
 * We use 32-bit pieces because these are the size of integers on which Javascript performs bit-operations. For operations like addition and multiplication, we split each number into 16-bit pieces, which can easily be multiplied within Javascript's floating-point representation without overflow and change in sign.
 * In the algorithms below, we frequently reduce the negative case to the positive case by negating the input(s) and then post-processing the result.
 * Note that we must ALWAYS check specially whether those values are MIN_VALUE = (-2^63) because -MIN_VALUE == MIN_VALUE (since 2^63 cannot be represented as a positive number, it overflows back into a negative).
 * Not handling this case would often result in infinite recursion.
 */

// NOTE: Common constant values ZERO, ONE, NEG_ONE, etc. are defined below the from* methods on which they depend.

/**
 * A cache of the Long representations of small integer values.
 * @type {!Object}
 * @private
 */
Kotlin.Long.IntCache_ = {};

/**
 * Returns a Long representing the given (32-bit) integer value.
 * @param {number} value The 32-bit integer in question.
 * @return {!Kotlin.Long} The corresponding Long value.
 */
Kotlin.Long.fromInt = function(value) {
  if (-128 <= value && value < 128) {
    var cachedObj = Kotlin.Long.IntCache_[value];
    if (cachedObj) {
      return cachedObj;
    }
  }

  var obj = new Kotlin.Long(value | 0, value < 0 ? -1 : 0);
  if (-128 <= value && value < 128) {
  }
  return obj;
};

/**
 * Returns a Long representation of the given string, written using the given radix.
 * @param {string} str The textual representation of the Long.
 * @param {number=} opt_radix The radix in which the text is written.
 * @return {!Kotlin.Long} The corresponding Long value.
 */
Kotlin.Long.fromString = function(str, opt_radix) {
  if (str.length == 0) {
    throw Error('number format error: empty string');
  }

  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  // Do several (8) digits each time through the loop, so as to minimize the calls to the very expensive emulated div.
  var radixToPower =
Kotlin.Long.fromString(Math.pow(radix, 8));
var result = Kotlin.Long.ZERO;
for (var i = 0; i < str.length; i += 8) {
  var size = Math.min(8, str.length - i);
  var value = parseInt(str.substring(i, i + size), radix);
  if (size < 8) {
    var power = Kotlin.Long.fromString(Math.pow(radix, size));
    result = result.multiply(power).add(Kotlin.Long.fromString(value));
  } else {
    result = result.multiply(radixToPower);
    result = result.add(Kotlin.Long.fromString(value));
  }
}
return result;

NOTE: the compiler should inline these constant values below and then remove/unwrap these variables, so there should be no runtime penalty for these. **Number used repeated below in calculations. This must appear before the `*` first call to any from* function below.**

```
*\@private
*\@type {number}
*Kotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;
*Kotlin.Long.TWO_PWR_24_DBL_ = 1 << 24;
*Kotlin.Long.TWO_PWR_32_DBL_ = Kotlin.Long.TWO_PWR_16_DBL_ * @type {number} n * @private
*Kotlin.Long.TWO_PWR_63_DBL_ = Kotlin.Long.TWO_PWR_64_DBL_ / 2;
*Kotlin.Long.ZERO = Kotlin.Long.fromString(0);
*Kotlin.Long.ONE = Kotlin.Long.fromString(1);
*Kotlin.Long.NEG_ONE = Kotlin.Long.fromInt(-1);
*Kotlin.Long.MAX_VALUE = Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);
*Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);
*Kotlin.Long.prototype.toInt = function() {
  return this.getNumber().toInt();
};
*Kotlin.Long.prototype.toString = function(opt_radix) {
  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  if (this.isZero()) {
    return '0';
  }

  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      // We need to change the Long value before it can be negated, so we remove the bottom-most digit in this base and then recurse to do the rest.
      var radixLong = Kotlin.Long.fromString(0);
      var rem = div.multiply(radixLong).subtract(this);
      return div.toString(radix) + rem.toInt().toString(radix);
    } else {
      return '-' + this.negate().toString(radix);
    }
  }

  // Do several (6) digits each time through the loop, so as to minimize the calls to the very expensive emulated div. var radixToPower = Kotlin.Long.fromString(Math.pow(radix, 6));
  var rem = this;
  var result = Kotlin.Long.ZERO;
  while (true) {
    var intval = rem.subtract(rem.multiply(radixToPower)).toInt();
    var digits = intval.toString(radix);
    var digits = '' + digits;
    var result = '' + digits + result;
    while (digits.length < 6) {
      digits = '' + digits;
      var result = '' + digits + result;
    }
    if (result.length > 0) {
      result = '' + digits + result;
    }
    rem = rem.subtract(digitToPower);
    if (result.length < 6) {
      result = result + '0';
    }
  }
  return result;
};
```
```javascript
Kotlin.Long.prototype.getNumBitsAbs = function() {
  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      return 64;
    } else {
      return this.negate().getNumBitsAbs();
    }
  } else {
    var val = this.high_ != 0 ? this.high_ : this.low_;
    for (var bit = 31; bit > 0; bit--) {
      if ((val & (1 << bit)) != 0) {
        break;
      }
    }
    return this.high_ != 0 ? bit + 33 : bit + 1;
  }
};

Kotlin.Long.prototype.isZero = function() {
  return this.high_ == 0 && this.low_ == 0;
};

Kotlin.Long.prototype.isNegative = function() {
  return this.high_ < 0;
};

Kotlin.Long.prototype.isOdd = function() {
  return (this.low_ & 1) == 1;
};

Kotlin.Long.prototype.equalsLong = function(other) {
  return (this.high_ == other.high_) && (this.low_ == other.low_);
};

Kotlin.Long.prototype.notEqualsLong = function(other) {
  return (this.high_ != other.high_) || (this.low_ != other.low_);
};

Kotlin.Long.prototype.lessThan = function(other) {
  return this.compare(other) < 0;
};

Kotlin.Long.prototype.lessThanOrEqual = function(other) {
  return this.compare(other) <= 0;
};

Kotlin.Long.prototype.greaterThan = function(other) {
  return this.compare(other) > 0;
};

Kotlin.Long.prototype.greaterThanOrEqual = function(other) {
  return this.compare(other) >= 0;
};

Kotlin.Long.prototype.compare = function(other) {
  if (this.equalsLong(other)) {
    return 0;
  }

  var thisNeg = this.isNegative();
  var otherNeg = other.isNegative();
  if (thisNeg && !otherNeg) {
    return -1;
  }
  if (!thisNeg && otherNeg) {
    return 1;
  }

  if (this.isZero() || other.isZero()) {
    return 0;
  }

  var one = Kotlin.Long.ONE;
  if (this.isNegative() && other.isNegative()) {
    return this.add(other.negate());
  } else if (this.isZero() || other.isNegative()) {
    return this.not().add(Kotlin.Long.ONE);
  } else {
    var thisNeg = this.isNegative();
    var oneNeg = one.isNegative();
    if (!thisNeg && !otherNeg) {
      return 0;
    }
    if (thisNeg) {
      return thisNeg ? this : this.negate();
    } else if (otherNeg) {
      return thisNeg ? other : other.negate();
    }
  }
};

Kotlin.Long.prototype.not = function() {
  return this.negate();
};

Kotlin.Long.prototype.add = function(other) {
  var a48 = this.high_ >> 16;
  var a32 = this.high_ & 0xFFFF;
  var a16 = this.low_ >> 16;
  var a00 = this.low_ & 0xFFFF;
  var b48 = other.high_ >> 16;
  var b32 = other.high_ & 0xFFFF;
  var b16 = other.low_ >> 16;
  var b00 = other.low_ & 0xFFFF;
  var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
  c00 += a00 + b00;
  c16 += c00 >> 16;
  c00 &= 0xFFFFFFFF;
  c16 += a16 + b16;
  c32 += c16 >> 16;
  c16 &= 0xFFFFFFFF;
  c32 += a32 + b32;
  c48 += c32 >> 16;
  c32 &= 0xFFFF;
  c48 += a48 + b48;
  c48 &= 0xFFFFFFFF;
  return Kotlin.Long.fromBits(c16 << 16 | c00, (c48 << 16) | c32);
};
```

Open Source Used In DNAC Platform Arica 7683
this.negate().multiply(other.negate())
else { return this.negate().multiply(other).negate(); }

// If both longs are small, use float multiplication
if (this.lessThan(Kotlin.Long.TWO_PWR_24_) &&
other.lessThan(Kotlin.Long.TWO_PWR_24_)) {
  return Kotlin.Long.fromNumber(this.toNumber() *
onother.toNumber());
}

// Divide each long into 4 chunks of 16 bits, and then add up 4x4 products.
if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
  if (other.equalsLong(Kotlin.Long.ONE) ||
onother.equalsLong(Kotlin.Long.NEG_ONE)) {
    return Kotlin.Long.MIN_VALUE;  // recall that -MIN_VALUE == MIN_VALUE
  } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {
    return Kotlin.Long.ONE;
  } else {
    // At this point, we have |other| >= 2, so [this/other] < [MIN_VALUE].
    var halfThis = this.shiftRight(1);
    var approx = halfThis.div(other).shiftLeft(1);
    if (approx.equalsLong(Kotlin.Long.ZERO)) {
      return other.isNegative() ? Kotlin.Long.ONE : Kotlin.Long.NEG_ONE;
    } else {
      var rem = this.subtract(other.multiply(approx));
      var result = approx.add(rem.div(other));
      return result;
    }
  }
}

// Repeat the following until the remainder is less than or equal to the real value so that the
// remainder never becomes negative.
var res = Kotlin.Long.ZERO;
var rem = this;
while (rem.greaterThanOrEqual(other)) {
  // Approximate the result of division. This may be a little greater or
  // smaller than the actual value.
  var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));
  // We will tweak the approximate result by changing it in the 48-th digit or
  // the smallest non-fractional digit, whichever is larger.
  var log2 = Math.ceil(Math.log(approx) / Math.LN2);
  var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);
  // Decrease the approximation until it is smaller than the remainder.
  // Note that if it is too large, the product overflows and is negative.
  var approxRem = approxRem.multiply(other);
  while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
    approx -= delta;
    approxRem = approxRem.multiply(other);
  }
  // We know the answer can't be zero... and actually, zero would cause
  // infinite recursion since we would make no progress.
  if (approxRem.isZero()) {
    return (rem & other) ? Kotlin.Long.ZERO : Kotlin.Long.ONE;
  } else if (remaining < [MIN_VALUE]) {
    var newRem = rem.subtract(other.multiply(approx));
    var result = approx.add(newRem.div(other));
    return result;
  }
  // So the remainder can be negative.
  var newRem = rem.subtract(other.multiply(approx));
  var result = approx.add(newRem.div(other));
  return result.
}

// This Long divided by the given one.
* @param {Kotlin.Long} other Long by which to divide.
* @return {Kotlin.Long} This Long divided by the given one.
*/
Kotlin.Long.prototype.div = function(other) {
  return this.div(other);
};

/**
 * Returns this Long modulo the given one.
 * @param {Kotlin.Long} other Long by which to mod.
 * @return {!Kotlin.Long} This Long modulo the given one.
 */
Kotlin.Long.prototype.modulo = function(other) {
  return this.modulo(other);
};

/**
 * Returns the bitwise-NOT of this value.
 * @return {!Kotlin.Long} The bitwise-NOT of this value.
 */
Kotlin.Long.prototype.not = function() {
  return this.not();
};

/**
 * Returns the bitwise-AND of this Long and the given one.
 * @param {Kotlin.Long} other The Long with which to AND.
 * @return {Kotlin.Long} The bitwise-AND of this and the other.
 */
Kotlin.Long.prototype.and = function(other) {
    return Kotlin.Long.fromBits(this.low_ & other.low_,
    this.high_ & other.high_);
};

Kotlin.Long.prototype.or = function(other) {
    return Kotlin.Long.fromBits(this.low_ | other.low_,
    this.high_ | other.high_);
};

Kotlin.Long.prototype.xor = function(other) {
    return Kotlin.Long.fromBits(this.low_ ^ other.low_,
    this.high_ ^ other.high_);
};

Kotlin.Long.prototype.shiftLeft = function(numBits) {
    numBits &= 63;
    if (numBits == 0) {
        return this;
    } else {
        var low = this.low_;
        if (numBits < 32) {
            var high = this.high_;
            return Kotlin.Long.fromBits(
                low << numBits,
                (high << numBits) | (low >>> (32 - numBits)));
        } else {
            return Kotlin.Long.fromBits(0, low << (numBits - 32));
        }
    }
};

Kotlin.Long.prototype.shiftRight = function(numBits) {
    numBits &= 63;
    if (numBits == 0) {
        return this;
    } else {
        var high = this.high_;
        if (numBits < 32) {
            var low = this.low_;
            return Kotlin.Long.fromBits(
                (low >>> numBits) | (high << (32 - numBits)),
                high >> numBits);
        } else if (numBits == 32) {
            return Kotlin.Long.fromBits(high, 0);
        } else {
            return Kotlin.Long.fromBits(high >> (numBits - 32), 0);
        }
    }
};

Kotlin.Long.prototype.shiftRightUnsigned = function(numBits) {
    numBits &= 63;
    if (numBits == 0) {
        return this;
    } else {
        var high = this.high_;
        if (numBits < 32) {
            var low = this.low_;
            return Kotlin.Long.fromBits(
                (low >>> numBits) | (high << (32 - numBits)),
                high >>> numBits);
        } else if (numBits == 32) {
            return Kotlin.Long.fromBits(high, 0);
        } else {
            return Kotlin.Long.fromBits(high >> (numBits - 32), 0);
        }
    }
};

Kotlin.defineModule = function (id, declaration) {
};

Kotlin.defineInlineFunction = function(tag, fun) {
    return fun;
};

Kotlin.wrapFunction = function(fun) {
    var f = function() {
        f = fun();
        return f.apply(this, arguments);
    };
    return function() {
        return f.apply(this, arguments);
    };
};

Kotlin.isTypeOf = function(type) {
    return function (object) {
        return typeof object === type;
    }
};

Kotlin.isInstanceOf = function (klass) {
    return function (object) {
        return Kotlin.isType(object, klass);
    }
};

Kotlin.orNull = function (fn) {
    return function (object) {
        return object == null ||
        fn(object);
    }
};

Kotlin.andPredicate = function (a, b) {
    return function (object) {
        return a(object) &&
        b(object);
    }
};

Kotlin.suspendCall = function(qualifier) {
    throwMarkerError();
};

Kotlin.coroutineResult = function(qualifier) {

    throwMarkerError();
};

Kotlin.coroutineController = function(qualifier) {

};
throwMarkerError();

Kotlin.coroutineReceiver = function(qualifier) {
    return function() {
        throw MarkerError();
    };
}

Kotlin.setCoroutineResult = function(value, qualifier) {
    return function() {
        throw MarkerError();
    };
}

Kotlin.getReifiedTypeParameterKType = function(typeParameter) {
    return function() {
        throw MarkerError();
    };
}

function throwMarkerError() {
    throw new Error("This marker function should never be called. Looks like compiler did not eliminate it properly. Please, report an issue if you caught this exception.");
}

Kotlin.getFunctionById = function(id, defaultValue) {
    return function() {
        return defaultValue;
    };
}

Kotlin.compareTo = function(a, b) {
    var typeA = typeof a;
    if (typeA === "number") {
        if (typeof b === "number") {
            return Kotlin.doubleCompareTo(a, b);
        }
        return Kotlin.primitiveCompareTo(a, b);
    }
    if (typeA === "string" || typeA === "boolean") {
        return Kotlin.primitiveCompareTo(a, b);
    }
    return a.compareTo_11rb$(b);
}

Kotlin.primitiveCompareTo = function(a, b) {
    return a < b ? -1 : a > b ? 1 : 0;
}

Kotlin.signBit = function(value) {
    return (value | 0) === value;
}

Kotlin.floatToRawBits = function(value) {
    var bufFloat32 = new Float32Array([value]);
    return bufInt32[0];
}

Kotlin.floatFromBits = function(value) {
    var bufInt32 = new Int32Array([value]);
    return bufFloat32[0];
}

Kotlin.doubleFromBits = function(value) {
    var bufInt64 = new Int64Array([value]);
    return bufFloat64[0];
}

Kotlin.numberHashCode = function(obj) {
    if ((obj | 0) === obj) {
        return obj | 0;
    } else {
        return (bufInt32[highIndex] * 31 | 0) + bufInt32[lowIndex] | 0;
    }
}

Kotlin.doubleToBits = function(value) {
    return Kotlin.doubleToRawBits(isNaN(value) ? NaN : value);
}

Kotlin.doubleToRawBits = function(value) {
    var bufFloat64 = new Float64Array([value]);
    return Kotlin.Long.fromBits(bufInt32[lowIndex], bufInt32[highIndex]);
}

Kotlin.floatToBits = function(value) {
    return Kotlin.floatToRawBits(isNaN(value) ? NaN : value);
}

Kotlin.floatToRawBits = function(value) {
    var bufFloat32 = new Float32Array([value]);
    return bufInt32[0];
}

Kotlin.doubleToRawBits = function(value) {
    var bufFloat64 = new Float64Array([value]);
    return bufInt32[0];
}

Kotlin.doubleSignBit = function(value) {
    return bufInt32[highIndex] & 0x80000000;
}

Kotlin.numberHashCode = function(obj) {
    if ((obj | 0) === obj) {
        return obj | 0;
    } else {
        return bufFloat64[0];
    }
}

Kotlin.floatToBits = function(value) {
    return bufFloat64[0];
}

Kotlin.doubleFromBits = function(value) {
    return bufFloat32[0];
}

Kotlin.doubleToBits = function(value) {
    return bufFloat64[0];
}

Kotlin.doubleToRawBits = function(value) {
    return bufInt32[0];
}

Kotlin.doubleFromBits = function(value) {
    return bufFloat32[0];
}

Kotlin.doubleToBits = function(value) {
    return bufFloat64[0];
}

Kotlin.doubleToRawBits = function(value) {
    return bufInt32[0];
}

if (typeof Math.sign === "undefined") {
    Math.sign = function(x) {
        x = +x; // convert to a number
        return x || x === 0 || isNaN(x) ? 1 : -1;
    };
}

if (typeof Math.trunc === "undefined") {
    Math.trunc = function(x) {
        return Number(x);
    };
}

Kotlin.ensureNotNull = function(x) {
    return x != null ? x : Kotlin.throwNPE();
}
var taylor_2_bound = Math.sqrt(epsilon);
var taylor_n_bound =
Math.sqrt(taylor_2_bound); var upper_taylor_2_bound =
1/taylor_2_bound; var upper_taylor_n_bound =
1/taylor_n_bound;

if (typeof Math.sinh === "undefined") {
    Math.sinh = function(x) {
        if (Math.abs(x) < taylor_n_bound) {
            var result = x;
        } else {
            var y = Math.exp(x);
            var y1 = 1 / y;
            if (isFinite(y)) return Math.exp(x - Math.LN2);
        }
    };
}

if (typeof Math.cosh === "undefined") {
    Math.cosh = function(x) {
        var y = Math.exp(x);
        var y1 = 1 / y;
        if (isFinite(y)) return Math.exp(x - Math.LN2);
    };
}

if (typeof Math.tanh === "undefined") {
    Math.tanh = function(x) {
        if (x >= +taylor_n_bound) {
            return Math.log((1 + x) / (1 - x)) / 2;
        } else if (x <= -taylor_n_bound) {
            // approximation by laurent series in 1/x at 0+ order from -1 to 0
            return Math.log(x + Math.LN2);
        } else {
            var result = x;
            if (Math.abs(x) >= taylor_2_bound) {
                var y3 = y * y;
                // approximation by taylor series in y at 0 up to order 2
                var result = x;
            } else {
                var y = Math.sqrt(2) * result;
            }
        }
    };
}

if (typeof Math.asinh === "undefined") {
    var asinh = function(x) {
        if (x >= +taylor_n_bound) {
            if (x > upper_taylor_n_bound)
            {
                return Math.log(x + Math.sqrt(x * x + 1));
            } else
            {
                // approximation by laurent series in 1/x at 0+ order from -1 to 0
                return Math.log(x) + Math.LN2;
            }
        } else
        {
            // approximation by taylor series in x at 0 up to order 4
            var result = x;
            if (Math.abs(x) >= taylor_2_bound) {
                var y3 = y * y;
                // approximation by taylor series in y at 0 up to order 2
                var result = x;
            } else {
                var result = x;
            }
        }
    };
}

if (typeof Math.acosh === "undefined") {
    Math.acosh = function(x) {
        if (x < 1)
        {
            return NaN;
        }
    };
}

if (typeof Math.asinh === "undefined") {
    var asinh = function(x) {
        if (x >= +taylor_n_bound) {
            if (x > upper_taylor_n_bound)
            {
                return Math.log(x + Math.sqrt(x * x + 1));
            } else
            {
                // approximation by laurent series in 1/x at 0+ order from -1 to 0
                return Math.log(x) + Math.LN2;
            }
        } else
        {
            // approximation by taylor series in x at 0 up to order 4
            var result = x;
            if (Math.abs(x) >= taylor_2_bound) {
                var y3 = y * y;
                // approximation by taylor series in y at 0 up to order 2
                var result = x;
            } else {
                var result = x;
            }
        }
    };
}

if (typeof Math.acosh === "undefined") {
    Math.acosh = function(x) {
        if (x < 1)
        {
            return NaN;
        }
    };
}

if (typeof Math.asinh === "undefined") {
    var asinh = function(x) {
        if (x >= +taylor_n_bound) {
            if (x > upper_taylor_n_bound)
            {
                return Math.log(x + Math.sqrt(x * x + 1));
            } else
            {
                // approximation by laurent series in 1/x at 0+ order from -1 to 0
                return Math.log(x) + Math.LN2;
            }
        } else
        {
            // approximation by taylor series in x at 0 up to order 4
            var result = x;
            if (Math.abs(x) >= taylor_2_bound) {
                var y3 = y * y;
                // approximation by taylor series in y at 0 up to order 2
                var result = x;
            } else {
                var result = x;
            }
        }
    };
}
Open Source Used In DNAC Platform Arica  7689

```javascript
value) {
    var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);
    if (propertyDescriptor != null &
```
@JsName("untypedCharArrayF")
inline fun untypedCharArrayWithFun(size: Int, init: (Int) -> Char): Array<Char> {
    val array = Array<Char>(size)
    for (i in 0..array.size - 1) {
        val value = init(i)
        js("array[i] = value;")
    }
    return array
}

@JsName("longArray")
fun longArray(size: Int, init: dynamic): Array<Long> {
    val result: dynamic = Array<Long>(size)
    result.`$type$` = "LongArray"
    return when (init) {
        null, true -> fillArrayVal(result, 0L)
        false -> result
        else -> fillArrayFun<Long>(result, init)
    }
}

@JsName("longArrayF")
inline fun longArrayWithFun(size: Int, init: (Int) -> Long): Array<Long> =
    fillArrayFun(longArray(size, false), init)

private fun <T> fillArrayVal(array: Array<T>, initValue: T): Array<T> {
    for (i in 0..array.size - 1) {
        array[i] = initValue
    }
    return array
}
get(0)\n\n/* Returns 2nd *element* from the array.\n  * If the size of this array is less than 2, throws an
  * [IndexOutOfBoundsException] except in Kotlin/JS\n  */
@kotlin.internal.InlineOnly\n@公开 inline operator fun <T> Array<out T>.component2(): T {
  return get(1)\n  /* Returns 2nd *element* from the array.\n   * If the size of this array is less than 2, throws an
   * [IndexOutOfBoundsException] except in Kotlin/JS\n   */
  @kotlin.internal.InlineOnly\n  @公开 inline operator fun ByteArray.component2(): Byte {
    return get(1)\n    /* Returns 2nd *element* from the array.\n      * If the size of this array is less than 2, throws an
      * [IndexOutOfBoundsException] except in Kotlin/JS\n      */
    @kotlin.internal.InlineOnly\n    @公开 inline operator fun ShortArray.component2(): Short {
      return get(1)\n      /* Returns 2nd *element* from the array.\n        * If the size of this array is less than 2, throws an
        * [IndexOutOfBoundsException] except in Kotlin/JS\n        */
        @kotlin.internal.InlineOnly\n        @公开 inline operator fun IntArray.component2(): Int {
          return get(1)\n          /* Returns 2nd *element* from the array.\n            * If the size of this array is less than 2, throws an
            * [IndexOutOfBoundsException] except in Kotlin/JS\n            */
            @kotlin.internal.InlineOnly\n            @公开 inline operator fun LongArray.component2(): Long {
              return get(1)\n              /* Returns 2nd *element* from the array.\n                * If the size of this array is less than 2, throws an
                * [IndexOutOfBoundsException] except in Kotlin/JS\n                */
                @kotlin.internal.InlineOnly\n                @公开 inline operator fun FloatArray.component2(): Float {
                  return get(1)\n                  /* Returns 2nd *element* from the array.\n                    * If the size of this array is less than 2, throws an
                    * [IndexOutOfBoundsException] except in Kotlin/JS\n                    */
                    @kotlin.internal.InlineOnly\n                    @公开 inline operator fun DoubleArray.component2(): Double {
                      return get(1)\n                      /* Returns 2nd *element* from the array.\n                        * If the size of this array is less than 2, throws an
                        * [IndexOutOfBoundsException] except in Kotlin/JS\n                        */
                        @kotlin.internal.InlineOnly\n                        @公开 inline operator fun BooleanArray.component2(): Boolean {
                          return get(1)\n                          /* Returns 2nd *element* from the array.\n                            * If the size of this array is less than 2, throws an
                            * [IndexOutOfBoundsException] except in Kotlin/JS\n                            */
                            @kotlin.internal.InlineOnly\n                            @公开 inline operator fun CharArray.component2(): Char {
                              return get(1)\n                              /* Returns 3rd *element* from the array.\n                                * If the size of this array is less than 3, throws an
                                * [IndexOutOfBoundsException] except in Kotlin/JS\n                                */
                                @kotlin.internal.InlineOnly\n                                @公开 inline operator fun ByteArray.component3(): Byte {
                                  return get(2)\n                                  /* Returns 3rd *element* from the array.\n                                    * If the size of this array is less than 3, throws an
                                    * [IndexOutOfBoundsException] except in Kotlin/JS\n                                    */
                                    @kotlin.internal.InlineOnly\n                                    @公开 inline operator fun ShortArray.component3(): Short {
                                      return get(2)\n                                      /* Returns 3rd *element* from the array.\n                                        * If the size of this array is less than 3, throws an
                                        * [IndexOutOfBoundsException] except in Kotlin/JS\n                                        */
                                        @kotlin.internal.InlineOnly\n                                        @公开 inline operator fun IntArray.component3(): Int {
                                          return get(2)\n                                          /* Returns 3rd *element* from the array.\n                                            * If the size of this array is less than 3, throws an
                                            * [IndexOutOfBoundsException] except in Kotlin/JS\n                                            */
                                            @kotlin.internal.InlineOnly\n                                            @公开 inline operator fun LongArray.component3(): Long {
                                              return get(2)\n                                              /* Returns 3rd *element* from the array.\n                                                * If the size of this array is less than 3, throws an
                                                * [IndexOutOfBoundsException] except in Kotlin/JS\n                                                */
                                                @kotlin.internal.InlineOnly\n                                                @公开 inline operator fun FloatArray.component3(): Float {
                                                  return get(2)\n                                                  /* Returns 3rd *element* from the array.\n                                                    * If the size of this array is less than 3, throws an
                                                    * [IndexOutOfBoundsException] except in Kotlin/JS\n                                                    */
                                                    @kotlin.internal.InlineOnly\n                                                    @公开 inline operator fun DoubleArray.component3(): Double {
                                                      return
get(2)\n\n/*\n * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component3(): Boolean {\n    return
}

/**\n * Returns 3rd *element* from the array.\n * If the size of this array is less than 3, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component3(): Char {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<T>.component4(): T {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component4(): Byte {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component4(): Short {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component4(): Int {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component4(): Long {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component4(): Boolean {\n    return
}

/**\n * Returns 4th *element* from the array.\n * If the size of this array is less than 4, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component4(): Char {\n    return
}

/**\n * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component5(): Short {\n    return
}

/**\n * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component5(): Int {\n    return
}

/**\n * Returns 5th *element* from the array.\n * If the size of this array is less than 5, throws an
 [IndexOutOfBoundsException] except in Kotlin/JS.\n */\n
@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component5(): Long {\n    return
}
get(4)

Returns 5th *element* from the array.

If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component5(): Float { return get(4)

Returns 5th *element* from the array.

If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component5(): Double { return get(4)

Returns 5th *element* from the array.

If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component5(): Boolean { return get(4)

Returns 5th *element* from the array.

If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.

@kotlin.internal.InlineOnly
public inline operator fun CharArray.component5(): Char { return get(4)

Returns `true` if [element] is found in the array.

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.contains(element: Byte): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.contains(element: Short): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun IntArray.contains(element: Int): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun LongArray.contains(element: Long): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.contains(element: Float): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.contains(element: Double): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.contains(element: Boolean): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun CharArray.contains(element: Char): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.contains(element: Byte): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.contains(element: Short): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun IntArray.contains(element: Int): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun LongArray.contains(element: Long): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.contains(element: Float): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.contains(element: Double): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.contains(element: Boolean): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun CharArray.contains(element: Char): Boolean { return indexOf(element) >= 0

public expect fun <T> Array<out T>.contains(element: T): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.contains(element: Byte): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.contains(element: Short): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun IntArray.contains(element: Int): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun LongArray.contains(element: Long): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.contains(element: Float): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.contains(element: Double): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.contains(element: Boolean): Boolean { return indexOf(element) >= 0

@kotlin.internal.InlineOnly
public inline operator fun CharArray.contains(element: Char): Boolean { return indexOf(element) >= 0

expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

expect fun <T> Array<out T>.elementAt(index: Int): T @sample samples.collections.Collections.Elements.elementAt* @sample expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt* @sample expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt* @sample expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt* @sample expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt* @sample expect fun <T> Array<out T>.elementAt(index: Int): T

Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt* @sample
samples.collections.Collections.Elements.elementAt

*\npublic expect fun FloatArray.elementAt(index: Int):
Float\n
*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n
* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun DoubleArray.elementAt(index: Int): Double\n
*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun BooleanArray.elementAt(index: Int):
Boolean\n
*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun CharArray.elementAt(index: Int):
Char\n
*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun ByteArray.elementAt(index: Int): Byte

*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun ShortArray.elementAt(index: Int): Short

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun IntArray.elementAt(index: Int): Int

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun LongArray.elementAt(index: Int): Long

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun FloatArray.elementAt(index: Int): Float

*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun DoubleArray.elementAt(index: Int): Double

*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun BooleanArray.elementAt(index: Int):
Boolean\n
*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun CharArray.elementAt(index: Int):
Char\n
*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun ByteArray.elementAt(index: Int): Byte

*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun ShortArray.elementAt(index: Int): Short

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun IntArray.elementAt(index: Int): Int

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections.Elements.elementAt

*\npublic expect fun LongArray.elementAt(index: Int): Long

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly\n\npublic inline fun \n<T> Array<out T>.elementType(index: Int): T

*\nReturns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly\n\npublic inline fun \n<T> Array<out T>.elementTypeOrNull(index: Int): T?

*\nReturns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly\n\npublic inline fun \n<T> Array<out T>.elementTypeOrElse(index: Int, defaultValue: (Int) -> T)

*\nReturns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n*\n* @sample samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly\n\npublic inline func samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly\n\npublic inline func samples.collections.Collections_ELEMENTS@kotlin.internal.InlineOnly
element at the given [index] or `null` if the [index] is out of bounds of this array.

```kotlin
ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}

ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}

IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}

LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}

FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}

DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}

BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}
```

Returns an element at the given [index] or ‘null’ if the [index] is out of bounds of this array.

```kotlin
ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
}

ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
}

IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
}

LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}

FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
}

DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}

BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {
    return firstOrNull(predicate)
}
```

Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

```kotlin
ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
}

ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
}

IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
}

LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}

FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
}

DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}

BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {
    return firstOrNull(predicate)
}
```
Returns the first element matching the given [predicate], or `null` if no such element was found.

```kotlin
CharArray.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
}
```

```kotlin
ArrayOfT.find(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
}
```

```kotlin
ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
}
```

```kotlin
ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
}
```

```kotlin
IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}
```

```kotlin
LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}
```

```kotlin
FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}
```

```kotlin
DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}
```

```kotlin
BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {
    return lastOrNull(predicate)
}
```

```kotlin
ByteArray.first(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
ShortArray.first(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
IntArray.first(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
LongArray.first(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
FloatArray.first(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
DoubleArray.first(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```kotlin
BooleanArray.first(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```
array is empty.

```kotlin
array is empty.
```
return null
* Returns the first element matching the given [predicate], or `null` if the element was not found.

```kotlin
public inline fun ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun BooleanArray.firstOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or `null` if the element was not found.
 */
public inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char? {
    for (element in this)
        if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```
CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ByteArray.getOrNull(index: Int): Byte? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ShortArray.getOrNull(index: Int): Short? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun IntArray.getOrNull(index: Int): Int? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun LongArray.getOrNull(index: Int): Long? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun FloatArray.getOrNull(index: Int): Float? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun DoubleArray.getOrNull(index: Int): Double? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun BooleanArray.getOrNull(index: Int): Boolean? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun CharArray.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun <T> Array<out T>.indexOf(element: T): Int {
    if (element == null) {
        for (index in indices) {
            if (this[index] == null) {
                return index
            }
        }
    } else {
        for (index in indices) {
            if (element == this[index]) {
                return index
            }
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun ByteArray.indexOf(element: Byte): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun ShortArray.indexOf(element: Short): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun IntArray.indexOf(element: Int): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun LongArray.indexOf(element: Long): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun FloatArray.indexOf(element: Float): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun DoubleArray.indexOf(element: Double): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun BooleanArray.indexOf(element: Boolean): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.indexOf(element: Char): Int {
    for (index in indices) {
        if (element == this[index]) {
            return index
        }
    }
    return -1
}
element matching the given [predicate], or -1 if the array does not contain such element.

*public inline fun LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*public inline fun FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*public inline fun DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*public inline fun BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*public inline fun CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*public inline fun <T> Array<out T>.last(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun ByteArray.last(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun ShortArray.last(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun IntArray.last(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun LongArray.last(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun FloatArray.last(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun DoubleArray.last(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun BooleanArray.last(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun CharArray.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

*public inline fun <T> Array<out T>.last(predicate: (T) -> Boolean): T {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun ByteArray.last(predicate: (Byte) -> Boolean): Byte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun ShortArray.last(predicate: (Short) -> Boolean): Short {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun IntArray.last(predicate: (Int) -> Boolean): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun LongArray.last(predicate: (Long) -> Boolean): Long {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun FloatArray.last(predicate: (Float) -> Boolean): Float {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun DoubleArray.last(predicate: (Double) -> Boolean): Double {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun BooleanArray.last(predicate: (Boolean) -> Boolean): Boolean {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

*public inline fun CharArray.last(predicate: (Char) -> Boolean): Char {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
if no such element is found.
*/
public inline fun IntArray.last(predicate: (Int) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public inline fun LongArray.last(predicate: (Long) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public inline fun FloatArray.last(predicate: (Float) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public inline fun DoubleArray.last(predicate: (Double) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public inline fun BooleanArray.last(predicate: (Boolean) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public inline fun CharArray.last(predicate: (Char) -> Boolean):
// 1
val element = this[index]
if (predicate(element)) return element
throw NoSuchElementException("Array contains no element matching the predicate.
*/
public fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {
// 1
if (element == null) {
// 1
val index = index
if (element == this[index]) {
// 1
return index
else {
// 1
val index = index
if (element == this[index]) {
// 1
return index

}
Open Source Used In DNAC Platform Arica 7703

```java
DoubleArray.lastIndexOf(element: Double): Int |n for (index in indices.reversed()) |n if (element ==
this[index]) |n return index
|n return -1
|n return -1

CharArray.lastIndexOf(element: Char): Int |n for (index in indices.reversed()) |n if (element == this[index]) |n return index
|n return -1
|n return -1

LongArray.lastIndexOf(element: Long): Long |n return if (isEmpty()) null else this.size
- 1 |n return -1
|n return -1

ByteArray.lastIndexOf(element: Byte): Byte |n return if (isEmpty()) null else this.size - 1 |n return -1
|n return -1
```

```java
BooleanArray.lastIndexOf(element: Boolean): Int |n for (index in indices.reversed()) |n if (element == this[index]) |n return index
|n return -1
|n return -1

DoubleArray.lastIndexOf(element: Double): Int |n for (index in indices.reversed()) |n if (element ==
this[index]) |n return index
|n return -1

ByteArray.lastIndexOf(element: Byte): Byte |n return if (isEmpty()) null else this.size - 1 |n return -1

IntArray.lastIndexOf(element: Int): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

CharArray.lastIndexOf(element: Char): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

BooleanArray.lastIndexOf(element: Boolean): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

DoubleArray.lastIndexOf(element: Double): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

ByteArray.lastIndexOf(element: Byte): Byte |n return if (isEmpty()) null else this.size - 1 |n return -1

IntArray.lastIndexOf(element: Int): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

CharArray.lastIndexOf(element: Char): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

BooleanArray.lastIndexOf(element: Boolean): Int |n return if (isEmpty()) null else this.size - 1 |n return -1
```

```java
CharArray.lastOrNull(predicate: (Char) -> Boolean): Char? {
    for (index in this.indices.reversed()) {
        val
        element = this[index]
        if (predicate(element)) return element
    }
    return null
}

BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (index in this.indices.reversed()) {
        val
        element = this[index]
        if (predicate(element)) return element
    }
    return null
}

DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {
    for (index in this.indices.reversed()) {
        val
        element = this[index]
        if (predicate(element)) return element
    }
    return null
}

ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (index in this.indices.reversed()) {
        val
        element = this[index]
        if (predicate(element)) return element
    }
    return null
}

Array<out T>.lastOrNull(predicate: (T) -> Boolean): T? {
    for (index in this.indices.reversed()) {
        val
        element = this[index]
        if (predicate(element)) return element
    }
    return null
}
```

```java
CharArray.lastIndexOf(element: Char): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

BooleanArray.lastIndexOf(element: Boolean): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

DoubleArray.lastIndexOf(element: Double): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

ByteArray.lastIndexOf(element: Byte): Byte |n return if (isEmpty()) null else this.size - 1 |n return -1

IntArray.lastIndexOf(element: Int): Int |n return if (isEmpty()) null else this.size - 1 |n return -1
```

```java
CharArray.lastIndexOf(element: Char): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

BooleanArray.lastIndexOf(element: Boolean): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

DoubleArray.lastIndexOf(element: Double): Int |n return if (isEmpty()) null else this.size - 1 |n return -1

ByteArray.lastIndexOf(element: Byte): Byte |n return if (isEmpty()) null else this.size - 1 |n return -1

IntArray.lastIndexOf(element: Int): Int |n return if (isEmpty()) null else this.size - 1 |n return -1
```
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ByteArray.random(): Byte {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ShortArray.random(): Short {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntArray.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongArray.random(): Long {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun FloatArray.random(): Float {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.random(): Double {
    return random(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun CharArray.random(): Char {
    return random(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun <T> Array<out T>.random(random: Random): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun ByteArray.random(random: Random): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun ShortArray.random(random: Random): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun IntArray.random(random: Random): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun LongArray.random(random: Random): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun FloatArray.random(random: Random): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
public fun DoubleArray.random(random: Random): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}
if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
fun CharArray.random(random: Random): Char {
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.4")
fun <T> Array<T>.randomOrNull(random: Random): T? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

```
randomness, or `null` if this array is empty.

```kotlin
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public fun LongArray.randomOrNull(random: Random): Long?
    if (isEmpty())
        return null
    return get(random.nextInt(size))

@SinceKotlin("1.4")
@ExperimentalStdlibApi
public fun FloatArray.randomOrNull(random: Random): Float?
    if (isEmpty())
        return null
    return get(random.nextInt(size))

@SinceKotlin("1.4")
@ExperimentalStdlibApi
public fun DoubleArray.randomOrNull(random: Random): Double?
    if (isEmpty())
        return null
    return get(random.nextInt(size))

@SinceKotlin("1.4")
@ExperimentalStdlibApi
public fun BooleanArray.randomOrNull(random: Random): Boolean?
    if (isEmpty())
        return null
    return get(random.nextInt(size))

@SinceKotlin("1.4")
@ExperimentalStdlibApi
public fun CharArray.randomOrNull(random: Random): Char?
    if (isEmpty())
        return null
    return get(random.nextInt(size))
```

```kotlin
public fun <T> Array<out T>.single(): T
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun ByteArray.single(): Byte
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun ShortArray.single(): Short
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun IntArray.single(): Int
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun LongArray.single(): Long
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun FloatArray.single(): Float
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }

public fun DoubleArray.single(): Double
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
```

```kotlin
public fun CharArray.single(): Char
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
```
@Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {
    var single: Byte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Byte
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {
    var single: Short? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Short
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Int
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Long
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Float
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Double
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one
 * matching element.
 */
public inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Boolean
}
element matching the predicate.

```kotlin
@Suppress("UNCHECKED_CAST")
return single as
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```

```kotlin
@ Suppress("UNCHECKED_CAST")
return single
```

```kotlin
Boolean
```
was not found or more than one element was found.

 */
public inline fun DoubleArray.singleOrNull(predicate: (Double) -> Boolean): Double? {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.
 */
public inline fun BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.
 */
public inline fun CharArray.singleOrNull(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> Array<out T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ByteArray.drop(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ShortArray.drop(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun IntArray.drop(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun LongArray.drop(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.drop(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.drop(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.drop(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun CharArray.drop(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}
n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop\n            * /\n
/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> List<T>.dropLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun ByteArray.dropLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun ShortArray.dropLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun IntArray.dropLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun LongArray.dropLast(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.dropLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.dropLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.dropLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun CharArray.dropLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n            * Returns a list containing all elements except last [n] elements.
            * @throws IllegalArgumentException if [n] is negative.
            * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropWhile
 */
public inline fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropWhile
 */
public inline fun ShortArray.dropLastWhile(predicate: (Short) -> Boolean): List<Short> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}
except last elements that satisfy the given [predicate].\n
* \n* \n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Int) -> Boolean): List<Int> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* \n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Long) -> Boolean): List<Long> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* \n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Float) -> Boolean): List<Float> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* \n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Double) -> Boolean): List<Double> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* \n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* \n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Char) -> Boolean): List<Char> \nfor (index in lastIndex downTo 0) \nif (!predicate(this[index])) \nreturn take(index + 1) \nelse return emptyList() \n* 
\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (T) -> Boolean): List<T> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Short) -> Boolean): List<Short> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Byte) -> Boolean): List<Byte> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Int) -> Boolean): List<Int> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Long) -> Boolean): List<Long> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Double) -> Boolean): List<Double> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Boolean) -> Boolean): List<Boolean> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (Char) -> Boolean): List<Char> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n* @sample samples.collections.Collections.Transformations.dropWhile(predicate: (T) -> Boolean): List<T> \nfor (item in this) \nif (yielding) \nlist.add(item) \nelse if (!predicate(item)) \n\n\n\n
FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * @sample collections.Collections.Transformations.drop
 */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @sample collections.Collections.Filtering.filter
 */
public inline fun BooleanArray.filter(predicate: (Boolean) -> Boolean): List<Boolean> {
    var yielding = false
    val list = ArrayList<Boolean>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample collections.Collections.Filtering.filterIndexed
 */
public inline fun ByteArray.filterIndexed(predicate: (Int, Byte) -> Boolean): List<Byte> {
    var yielding = false
    val list = ArrayList<Byte>()
    for (index in this)
        if (yielding)
            list.add(index)
        else if (!predicate(index)) {
            list.add(index)
            yielding = true
        }
    return list
}

filterIndexedTo(ArrayList<Byte>(), predicate)\n@\n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/\npublic inline fun ShortArray.filterIndexed(predicate: (index: Int, Short) -> Boolean): List<Short> \n{ return filterIndexedTo(ArrayList<Short>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun IntArray.filterIndexed(predicate: (index: Int, Int) -> Boolean): List<Int> \n{ return filterIndexedTo(ArrayList<Int>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun LongArray.filterIndexed(predicate: (index: Int, Long) -> Boolean): List<Long> \n{ return filterIndexedTo(ArrayList<Long>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float> \n{ return filterIndexedTo(ArrayList<Float>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> \n{ return filterIndexedTo(ArrayList<Double>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> \n{ return filterIndexedTo(ArrayList<Boolean>(), predicate)\n} \n**\n* Returns a list containing only elements matching the given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexed\n*/
public inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> \n{ return filterIndexedTo(ArrayList<Char>(), predicate)\n}
/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Int>> IntArray.filterIndexedTo(destination: C, predicate: (index: Int, Int) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Long>> LongArray.filterIndexedTo(destination: C, predicate: (index: Int, Long) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Float>> FloatArray.filterIndexedTo(destination: C, predicate: (index: Int, Float) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Double>> DoubleArray.filterIndexedTo(destination: C, predicate: (index: Int, Double) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterIndexedTo(destination: C, predicate: (index: Int, Boolean) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */
@public inline fun <C : MutableCollection<in Char>> CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C { 
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @sample samples.collections.Collections.Filtering.filterIsInstanceTo
 */
@public inline fun <R> Array<*>.filterIsInstanceTo<R>(): List<R> {
    return filterIsInstanceTo(ArrayList<R>(), R)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filterNot
 */
@public inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {
    return filterNotTo(ArrayList<Byte>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filterNot
 */
@public inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {
    return filterNotTo(ArrayList<Short>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filterNot
 */
@public inline fun <R> Array<*>.filterNotTo<R>(): List<R> {
    return filterNotTo(ArrayList<R>(), R)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filterNot
 */
@public inline fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {
    return filterNotTo(ArrayList<T>(), predicate)
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @sample samples.collections.Collections.Filtering.filterIsInstanceTo
 */
@public inline fun <R> Array<*>.filterIsInstanceTo<R>(): List<R> {
    return filterIsInstanceTo(ArrayList<R>(), R)
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @sample samples.collections.Collections.Filtering.filterIsInstanceTo
 */
@public inline fun <R> Array<*>.filterIsInstanceTo<R>(predicate: (R) -> Boolean): List<R> {
    return filterIsInstanceTo(ArrayList<R>(), predicate)
}
filterNotTo(ArrayList<Short>(), predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the
given [predicate].\n * \n * @sample samples.collections.Collections.Filtering.filter\n */\npublic inline fun
IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {\n return filterNotTo(ArrayList<Int>(),
predicate)\n}\n\n/**\n * Returns a list containing all elements not matching the given [predicate].\n * \n * @sample
samples.collections.Collections.Filtering.filter\n */\npublic inline fun LongArray.filterNot(predicate: (Long) ->
Boolean): List<Long> {\n return filterNotTo(ArrayList<Long>(), predicate)\n}\n\n/**\n * Returns a list
containing all elements not matching the given [predicate].\n * \n * @sample
samples.collections.Collections.Filtering.filter\n */\npublic inline fun FloatArray.filterNot(predicate: (Float) ->
Boolean): List<Float> {\n return filterNotTo(ArrayList<Float>(), predicate)\n}\n\n/**\n * Returns a list
containing all elements not matching the given [predicate].\n * \n * @sample
samples.collections.Collections.Filtering.filter\n */\npublic inline fun DoubleArray.filterNot(predicate: (Double) ->
Boolean): List<Double> {\n return filterNotTo(ArrayList<Double>(), predicate)\n}\n\n/**\n * Returns a list
containing all elements not matching the given [predicate].\n * \n * @sample
samples.collections.Collections.Filtering.filter\n */\npublic inline fun BooleanArray.filterNot(predicate: (Boolean) > Boolean): List<Boolean> {\n return filterNotTo(ArrayList<Boolean>(), predicate)\n}\n\n/**\n * Returns a list
containing all elements not matching the given [predicate].\n * \n * @sample
samples.collections.Collections.Filtering.filter\n */\npublic inline fun CharArray.filterNot(predicate: (Char) ->
Boolean): List<Char> {\n return filterNotTo(ArrayList<Char>(), predicate)\n}\n\n/**\n * Returns a list containing
all elements that are not `null`.\n * \n * @sample samples.collections.Collections.Filtering.filterNotNull\n */\npublic
fun <T : Any> Array<out T?>.filterNotNull(): List<T> {\n return filterNotNullTo(ArrayList<T>())\n}\n\n/**\n *
Appends all elements that are not `null` to the given [destination].\n * \n * @sample
samples.collections.Collections.Filtering.filterNotNullTo\n */\npublic fun <C : MutableCollection<in T>, T : Any>
Array<out T?>.filterNotNullTo(destination: C): C {\n for (element in this) if (element != null)
destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not matching the given
[predicate] to the given [destination].\n * \n * @sample samples.collections.Collections.Filtering.filterTo\n
*/\npublic inline fun <T, C : MutableCollection<in T>> Array<out T>.filterNotTo(destination: C, predicate: (T) ->
Boolean): C {\n for (element in this) if (!predicate(element)) destination.add(element)\n return
destination\n}\n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n * \n *
@sample samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in
Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {\n for (element in this) if
(!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n * \n * @sample
samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Short>>
ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {\n for (element in this) if
(!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n * \n * @sample
samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Int>>
IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {\n for (element in this) if
(!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n * \n * @sample
samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Long>>
LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {\n for (element in this) if
(!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n * \n * @sample
samples.collections.Collections.Filtering.filterTo\n */\npublic inline fun <C : MutableCollection<in Float>>
FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {\n for (element in this) if
(!predicate(element)) destination.add(element)\n return destination\n}\n\n/**\n * Appends all elements not
matching the given [predicate] to the given [destination].\n * \n * @sample

Open Source Used In DNAC Platform Arica 7715


samples.collections.Collections.Filtering.filterTo\n *\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Double}\rangle) DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {\n    for (element in this) if (!predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Boolean}\rangle) BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {\n    for (element in this) if (!predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Char}\rangle) CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {\n    for (element in this) if (!predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} T, C : \text{MutableCollection<in \text{Array<out T>}}) Array<out T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Byte}\rangle) ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Short}\rangle) ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Int}\rangle) IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Long}\rangle) LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Float}\rangle) FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Double}\rangle) DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} C : \text{MutableCollection<in Char}\rangle) CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n}\n**\n*\nUBLIC inline fun \(\text{\textless} T, C : \text{MutableCollection<in \text{Array<out T>}}) Array<out T>.slice(indices: IntRange): List<T> \{\n    if (indices.isEmpty()) return listOf()\n    return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n**\n*\nUBLIC inline fun ByteArray.slice(indices: IntRange): List<Byte> \{\n    if (indices.isEmpty()) return listOf()\n    return copyOfRange(indices.start, indices.endInclusive + 1).asList()\n}\n   
Open Source Used In DNAC Platform Arica  7716
fun ShortArray.slice(indices: IntRange): List<Short> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun CharArray.slice(indices: IntRange): List<Char>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun IntArray.slice(indices: IntRange): List<Int>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun LongArray.slice(indices: IntRange): List<Long>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun FloatArray.slice(indices: IntRange): List<Float>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun DoubleArray.slice(indices: IntRange): List<Double>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun BooleanArray.slice(indices: IntRange): List<Boolean>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun ByteArray.slice(indices: IntRange): List<Byte>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

public fun <T> Array<out T>.slice(indices: IntRange): List<T>
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

Returns a list containing elements at indices in the specified [indices] range.

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns a list containing elements at indices in the specified [indices] range.

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun CharArray.slice(indices: IntRange): List<Char> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

Returns a list containing elements at indices in the specified [indices] range.

/**
 * Returns a list containing elements at specified [indices].
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Returns an array containing elements of this array at specified [indices].

/**
 * Returns an array containing elements at specified [indices].
 */
public fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}
ByteArray.sliceArray(indices: Collection<Int>): ByteArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ShortArray.sliceArray(indices: Collection<Int>): ShortArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

IntArray.sliceArray(indices: Collection<Int>): IntArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

LongArray.sliceArray(indices: Collection<Int>): LongArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

FloatArray.sliceArray(indices: Collection<Int>): FloatArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

CharArray.sliceArray(indices: Collection<Int>): CharArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ByteArray.sliceArray(indices: Collection<Int>, IntRange): ByteArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ShortArray.sliceArray(indices: Collection<Int>, IntRange): ShortArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

IntArray.sliceArray(indices: Collection<Int>, IntRange): IntArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

LongArray.sliceArray(indices: Collection<Int>, IntRange): LongArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

FloatArray.sliceArray(indices: Collection<Int>, IntRange): FloatArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

CharArray.sliceArray(indices: Collection<Int>, IntRange): CharArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ByteArray.sliceArray(indices: Collection<Int>, IntRange): ByteArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ShortArray.sliceArray(indices: Collection<Int>, IntRange): ShortArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

IntArray.sliceArray(indices: Collection<Int>, IntRange): IntArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

LongArray.sliceArray(indices: Collection<Int>, IntRange): LongArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

FloatArray.sliceArray(indices: Collection<Int>, IntRange): FloatArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

CharArray.sliceArray(indices: Collection<Int>, IntRange): CharArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ByteArray.sliceArray(indices: Collection<Int>, IntRange): ByteArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ShortArray.sliceArray(indices: Collection<Int>, IntRange): ShortArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

IntArray.sliceArray(indices: Collection<Int>, IntRange): IntArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

LongArray.sliceArray(indices: Collection<Int>, IntRange): LongArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

FloatArray.sliceArray(indices: Collection<Int>, IntRange): FloatArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

CharArray.sliceArray(indices: Collection<Int>, IntRange): CharArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ByteArray.sliceArray(indices: Collection<Int>, IntRange): ByteArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

ShortArray.sliceArray(indices: Collection<Int>, IntRange): ShortArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

IntArray.sliceArray(indices: Collection<Int>, IntRange): IntArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

LongArray.sliceArray(indices: Collection<Int>, IntRange): LongArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

FloatArray.sliceArray(indices: Collection<Int>, IntRange): FloatArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

CharArray.sliceArray(indices: Collection<Int>, IntRange): CharArray {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}
require(n >= 0) { 
    \"Requested element count \$n is less than zero.\" \\
} if (n == 0) return emptyList() \\
if (n > size) return toList() \\
if (n == 1) return listOf((this[0])[n]) \\
var count = 0 \\
val list = ArrayList<T>(n) \\
for (item in this) \\
    list.add(item) \\
    if (++count == n) \\
        break \\
} return list

/*
 * Returns a list containing first \[n\] elements.
 * @throws IllegalArgumentException if \[n\] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> ArrayList<T>.take(n: Int): List<T> \\
    require(n >= 0) { 
        \"Requested element count \$n is less than zero.\" \\
    } if (n == 0) return emptyList() \\
    if (n > size) return toList() \\
    if (n == 1) return listOf((this[0])[n]) \\
    var count = 0 \\
    val list = ArrayList<T>(n) \\
    for (item in this) \\
        list.add(item) \\
        if (++count == n) \\
            break \\
    } return list

/**
 * Returns a list containing last \[n\] elements.
 * @throws IllegalArgumentException if \[n\] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> ByteArray.take(n: Int): List<T> \\
    require(n >= 0) { 
        \"Requested element count \$n is less than zero.\" \\
    } if (n == 0) return emptyList() \\
    if (n > size) return toList() \\
    if (n == 1) return listOf((this[0])[n]) \\
    var count = 0 \\
    val list = ArrayList<T>(n) \\
    for (item in this) \\
        list.add(item) \\
        if (++count == n) \\
            break \\
    } return list

/**
 * Returns a list containing first \[n\] elements.
 * @throws IllegalArgumentException if \[n\] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> CharArray.take(n: Int): List<T> \\
    require(n >= 0) { 
        \"Requested element count \$n is less than zero.\" \\
    } if (n == 0) return emptyList() \\
    if (n > size) return toList() \\
    if (n == 1) return listOf((this[0])[n]) \\
    var count = 0 \\
    val list = ArrayList<T>(n) \\
    for (item in this) \\
        list.add(item) \\
        if (++count == n) \\
            break \\
    } return list

/**
 * Returns a list containing first \[n\] elements.
 * @throws IllegalArgumentException if \[n\] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> DoubleArray.take(n: Int): List<T> \\
    require(n >= 0) { 
        \"Requested element count \$n is less than zero.\" \\
    } if (n == 0) return emptyList() \\
    if (n > size) return toList() \\
    if (n == 1) return listOf((this[0])[n]) \\
    var count = 0 \\
    val list = ArrayList<T>(n) \\
    for (item in this) \\
ArrayList<T>(n)\n    for (index in size - n until size)\n        list.add(this[index])\n    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun <T> ArrayList<T>(n): List<T> {
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun List<T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun CharArray.takeLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Char>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun Array<out Char>.takeLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Char>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun ShortArray.takeLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Short>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun FloatArray.takeLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Float>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun LongArray.takeLast(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Long>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun IntArray.takeLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Int>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun ShortArray.takeLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Short>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun ByteArray.takeLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Byte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun BooleanArray.takeLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Boolean>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list\n\n/**\n * Returns a list containing last \[n\] elements.\n * @throws IllegalArgumentException if \[n\] is negative.\n * @sample samples.collections.Collections.Transformations.takeLast(n)\n */
public fun DoubleArray.takeLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size - n
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<Double>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
\n*/

```kotlin
return drop(index + 1) |\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n    return toList(|\n    ) |\n```
IllegalArgumentException if [fromIndex] is greater than [toIndex].

public fun ByteArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public fun ShortArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public fun IntArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public fun LongArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public fun FloatArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public fun DoubleArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
    }
}
IllegalArgumentException if [fromIndex] is greater than [toListIndex].

BooleanArray.reverse(fromIndex: Int, toIndex: Int): Unit

AbstractList.checkRangeIndexes(fromIndex, toIndex, size)

val midPoint = (fromIndex + toIndex) / 2

if (fromIndex == midPoint) return

var reverseIndex = toIndex - 1

for (index in fromIndex until midPoint) {
    val tmp = this[index]
    this[index] = this[reverseIndex]
    this[reverseIndex] = tmp
    reverseIndex--
}

return this

}

ShortArray.reverse(fromIndex: Int, toIndex: Int): Unit

AbstractList.checkRangeIndexes(fromIndex, toIndex, size)

val midPoint = (fromIndex + toIndex) / 2

if (fromIndex == midPoint) return

var reverseIndex = toIndex - 1

for (index in fromIndex until midPoint) {
    val tmp = this[index]
    this[index] = this[reverseIndex]
    this[reverseIndex] = tmp
    reverseIndex--
}

return this

}

ByteArray.reverse(fromIndex: Int, toIndex: Int): Unit

AbstractList.checkRangeIndexes(fromIndex, toIndex, size)

val midPoint = (fromIndex + toIndex) / 2

if (fromIndex == midPoint) return

var reverseIndex = toIndex - 1

for (index in fromIndex until midPoint) {
    val tmp = this[index]
    this[index] = this[reverseIndex]
    this[reverseIndex] = tmp
    reverseIndex--
}

return this

}

CharArray.reverse(fromIndex: Int, toIndex: Int): List<Char>


toMutableList()

list.reverse()

return list

}

BooleanArray.reverse(): List<Boolean>


toMutableList()

list.reverse()

return list

}

List<Char>


"1.4"

* Reverses elements of the array in the specified range in-place.

* @param fromIndex the start of the range (inclusive) to reverse.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toListIndex] is greater than the size of this array.

*

* * An public function.

ByteArray.reverse(): List<Byte>


"1.4"

* Returns a list with elements from reversed order.

* * An public function.

ByteArray.reverse(): List<Byte>


"1.4"

* Returns a list with elements from reversed order.

* * An public function.

IntArray.reverse(): List<Int>


"1.4"

* Returns a list with elements from reversed order.

* * An public function.

CharArray.reverse(): List<Char>


"1.4"

* Returns a list with elements from reversed order.

* * An public function.

ByteArray.reverse(): List<Byte>


"1.4"

* Returns a list with elements from reversed order.

* * An public function.

ArrayOfNulls(list, size)


"1.4"

* Returns a list with elements of this array in reversed order.

* * An public function.

ByteArray.reverseArray(): ByteArray


"1.4"

* Returns an array with elements of this array in reversed order.

* * An public function.

CharArray.reverseArray(): CharArray


"1.4"

* Returns an array with elements of this array in reversed order.

* * An public function.

ByteArray.reverseArray(): ByteArray


"1.4"

* Returns an array with elements of this array in reversed order.

* * An public function.

IntArray.reverseArray(): IntArray


"1.4"

* Returns an array with elements of this array in reversed order.

* * An public function.

ArrayOfNulls(list)
this
val result = DoubleArray(size)
val lastIndex = lastIndex
for (i in 0..lastIndex)
    result[lastIndex - i] = this[i]
return result

* */public fun BooleanArray.reversedArray(): BooleanArray
    { if (isEmpty()) return this
    val result = BooleanArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

* */public fun CharArray.reversedArray(): CharArray
    { if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

* */public fun<
    T>
Array<T>.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun ByteArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun ShortArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun IntArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun LongArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun FloatArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun DoubleArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun BooleanArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun CharArray.shuffle(): Unit
    { shuffle(Random)
    }

* */public fun Array<T>.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun ByteArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun ShortArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun IntArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun LongArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun FloatArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun DoubleArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun BooleanArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */public fun CharArray.shuffle(random: Random): Unit
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }

* */
/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun FloatArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun DoubleArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun BooleanArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun CharArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun ByteArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun ShortArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun IntArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun LongArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun FloatArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun DoubleArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun CharArray.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun ByteArray.sorted(): List<Byte> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun ShortArray.sorted(): List<Short> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun IntArray.sorted(): List<Int> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun LongArray.sorted(): List<Long> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun FloatArray.sorted(): List<Float> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun DoubleArray.sorted(): List<Double> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.4")
public fun CharArray.sorted(): List<Char> {
    return toTypedArray().apply { sort() }.asList()
}
elements sorted according to their natural sort order.

```kotlin
public fun LongArray.sorted(): List<Long> {
    return toTypedArray().apply { sort() } asList()
}

public fun FloatArray.sorted(): List<Float> {
    return toTypedArray().apply { sort() } asList()
}

public fun DoubleArray.sorted(): List<Double> {
    return toTypedArray().apply { sort() } asList()
}

public fun CharArray.sorted(): List<Char> {
    return toTypedArray().apply { sort() } asList()
}
```

```kotlin
public fun <T : Comparable<T>> Array<T>.sortedArray(): Array<T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun ByteArray.sortedArray(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun ShortArray.sortedArray(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun IntArray.sortedArray(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun LongArray.sortedArray(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun FloatArray.sortedArray(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

public fun DoubleArray.sortedArray(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun CharArray.sortedArray(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T> Array<out T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(comparator) }
}
```

```kotlin
Array<T>.sortedArrayDescending(): Array<T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}
```

```kotlin
ByteArray.sortedArrayDescending(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
ShortArray.sortedArrayDescending(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
IntArray.sortedArrayDescending(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
LongArray.sortedArrayDescending(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
DoubleArray.sortedArrayDescending(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

```kotlin
Array<T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(comparator) }
}
```

```kotlin
Array<out T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(comparator) }
}
```
of all elements sorted according to natural sort order of the value returned by specified [selector] function.

The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

* @sample samples.collections.Collections.Sorting.sortedBy
 */

```kotlin
public inline fun <T, R : Comparable<R>>
Array<out T>.sortBy(crossinline selector: (T) -> R?): List<T> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
BooleanArray.sortedBy(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {
    return
    sortedWith(compareBy(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 * Note: The sort is _stable_. It preserves the order relative to each other after sorting.
 * @sample samples.collections.Collections.Sorting.sortedByDescending
 */
public inline fun <T, R : Comparable<R>>
Array<out T>.sortByDescending(crossinline selector: (T) -> R?): List<T> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
ShortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
BooleanArray.sortedByDescending(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

/**
 * Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.
 */
public inline fun <R : Comparable<R>>
CharArray.sortedByDescending(crossinline selector: (Char) -> R?): List<Char> {
    return
    sortedWith(compareByDescending(selector))
}/*
 */

```
to the specified [comparator].

```kotlin
/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
private fun Array<out T>.asList(): List<T> = List(this)
```

```kotlin
/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepHashCode(): Int
```

```kotlin
/**
 * Returns a string representation of the contents of this array as if it is a
 * [List].
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepToString(): String
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. If two corresponding elements are nested arrays, they are also compared deeply. If any of arrays contains itself on any nesting level the behavior is undefined. The elements of other types are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. `null` is equal to `null`. If any of arrays contains itself on any nesting level the behavior is undefined.
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepEquals(other: Array<out T>?): Boolean
```

```kotlin
/**
 * Returns a hash code based on the contents of this array as if it is a [List].
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepHashCode(): Int
```

```kotlin
/**
 * Returns a string representation of the contents of this array as if it is a
 * [List].
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepToString(): String
```

```kotlin
/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepHashCode(): Int
```

```kotlin
/**
 * Returns a string representation of the contents of this array as if it is a
 * [List].
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public expect fun <T> Array<out T>?.contentDeepToString(): String
```
expect infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun ByteArray.contentEquals(other: ByteArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun ShortArray.contentEquals(other: ShortArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun IntArray.contentEquals(other: IntArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun LongArray.contentEquals(other: LongArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, &i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun CharArray.contentEquals(other: CharArray): Boolean

expect infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean
two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
@SinceKotlin("1.4")
public expect infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the \[equals\][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.

```
Open Source Used In DNAC Platform Arica 7733
FloatArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun DoubleArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun BooleanArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect fun CharArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun <T> Array<out T>?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun ByteArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun ShortArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun IntArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun LongArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun FloatArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun DoubleArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun BooleanArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
public expect fun CharArray?.contentToString(): String

/**
 * Copies this array or its subrange into the destination array and returns that array.
 * It's allowed to pass the same array in the destination and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the destination array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array or when `startIndex > endIndex`
 * @return the destination array.
 */
@SinceKotlin("1.3")
public expect fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T>

Open Source Used In DNAC Platform Arica  7734
the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @returns the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.\n * It's allowed to pass the same array in the [destination] array and returns that array.\n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @returns the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.\n * It's allowed to pass the same array in the [destination] array and returns that array.\n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @returns the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.\n * It's allowed to pass the same array in the [destination] array and returns that array.\n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @returns the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.\n * It's allowed to pass the same array in the [destination] array and returns that array.\n * @param destination the array to copy to.\n * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @returns the [destination] array.
 */
Open Source Used In DNAC Platform Arica 7736

*/
@SinceKotlin("1.3")
public expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException or [IllegalStateException] when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException or [IllegalStateException] when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
public expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOf(): Array<T>

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun ByteArray.copyOf(): ByteArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun ShortArray.copyOf(): ShortArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun IntArray.copyOf(): IntArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun LongArray.copyOf(): LongArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun FloatArray.copyOf(): FloatArray

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun DoubleArray.copyOf():
DoubleArray

* Returns new array which is a copy of the original array.

public expect fun BooleanArray.copyOf():

BooleanArray

* Returns new array which is a copy of the original array.

public expect fun CharArray.copyOf():

CharArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample

samples.collections.Arrays.CopyOfOperations.copyOf

* @sample

samples.collections.Arrays.CopyOfOperations.copyOf:

CharArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @sample

ByteArray.copyOf

* Copies the given ByteArray object.

 public expect fun ByteArray.copyOf(newSize: Int): ByteArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @sample

IntArray.copyOf

* Copies the given IntArray object.

 public expect fun IntArray.copyOf(newSize: Int): IntArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @sample

CharArray.copyOf

* Copies the given CharArray object.

 public expect fun CharArray.copyOf(newSize: Int): CharArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with null char (’\u0000’) values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (’\u0000’) values.

* @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @sample

BooleanArray.copyOf

* Copies the given BooleanArray object.

 public expect fun BooleanArray.copyOf(newSize: Int): BooleanArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with null char (’\u0000’) values if necessary.

* * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (’\u0000’) values.

* @sample

samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @sample

CharArray.copyOf

* Copies the given CharArray object.

 public expect fun CharArray.copyOf(newSize: Int): CharArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with null values if
necessary. If [newSize] is less than the size of the original array, the copy array is truncated to the
[newSize]. If [newSize] is greater than the size of the original array, the extra elements in the copy array are
filled with `null` values.

* - If [newSize] is less than zero or [toIndex] is greater than the size of this array, @throws
IllegalArgumentOutOfRangeException if [fromIndex] is greater than [toIndex].

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than
the size of this array.
CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit
IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for
 * an IntArray.
 */
public val IntArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for
 * a ShortArray.
 */
public val ShortArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for
 * a LongArray.
 */
public val LongArray.indices: IntRange
    get() = IntRange(0, lastIndex)

 /**
 * Returns the range of valid indices for
 * a CharArray.
 */
public val CharArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val ByteArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val ShortArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val IntArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val LongArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val FloatArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val DoubleArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the
 * array.
 */
@kotlin.internal.InlineOnly
public val BooleanArray.lastIndex: Int
    get() = size - 1

/**
 * Returns an array containing all
 * elements of the original array and
 * then the given [element].
 */

Open Source Used in DNAC Platform Arica 7740
public expect operator fun <T> Array<T>.plus(element: T): Array<T> Returns an array containing all elements of the original array and then the given [element].

public expect operator fun ByteArray.plus(element: Byte): ByteArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun ShortArray.plus(element: Short): ShortArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun IntArray.plus(element: Int): IntArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun LongArray.plus(element: Long): LongArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun FloatArray.plus(element: Float): FloatArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun DoubleArray.plus(element: Double): DoubleArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun BooleanArray.plus(element: Boolean): BooleanArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun CharArray.plus(element: Char): CharArray Returns an array containing all elements of the original array and then the given [element].

public expect operator fun Collection<T>.plus(elements: Collection<T>): Collection<T> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Byte>.plus(elements: Collection<Byte>): Collection<Byte> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Short>.plus(elements: Collection<Short>): Collection<Short> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Int>.plus(elements: Collection<Int>): Collection<Int> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Long>.plus(elements: Collection<Long>): Collection<Long> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Float>.plus(elements: Collection<Float>): Collection<Float> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Double>.plus(elements: Collection<Double>): Collection<Double> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Boolean>.plus(elements: Collection<Boolean>): Collection<Boolean> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Char>.plus(elements: Collection<Char>): Collection<Char> Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

public expect operator fun Collection<Array<out T>>.plus(elements: Array<out T>): Array<out T> Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun ByteArray.plus(elements: ByteArray): ByteArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun ShortArray.plus(elements: ShortArray): ShortArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun IntArray.plus(elements: IntArray): IntArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun LongArray.plus(elements: LongArray): LongArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun FloatArray.plus(elements: FloatArray): FloatArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.

public expect operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray Returns an array containing all elements of the original array and then all elements of the given [elements] array.
BooleanArray.plus(elements: BooleanArray): BooleanArray\n\n/**\n * Returns an array containing all elements of
the original array and then all elements of the given [elements] array.\n */\npublic expect operator fun
CharArray.plus(elements: CharArray): CharArray\n\n/**\n * Returns an array containing all elements of the original
array and then the given [element].\n */\n@Suppress(\"NO_ACTUAL_FOR_EXPECT\")\npublic expect fun <T>
Array<T>.plusElement(element: T): Array<T>\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun IntArray.sort(): Unit\n\n/**\n * Sorts the array
in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun LongArray.sort():
Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic
expect fun ByteArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun ShortArray.sort(): Unit\n\n/**\n * Sorts the
array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun
DoubleArray.sort(): Unit\n\n/**\n * Sorts the array in-place.\n * \n * @sample
samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun FloatArray.sort(): Unit\n\n/**\n * Sorts the
array in-place.\n * \n * @sample samples.collections.Arrays.Sorting.sortArray\n */\npublic expect fun
CharArray.sort(): Unit\n\n/**\n * Sorts the array in-place according to the natural order of its elements.\n * \n * The
sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n *
@sample samples.collections.Arrays.Sorting.sortArrayOfComparable\n */\npublic expect fun <T :
Comparable<T>> Array<out T>.sort(): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n * The sort is
_stable_. It means that equal elements preserve their order relative to each other after sorting.\n * \n * @param
fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range
(exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is
less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if
[fromIndex] is greater than [toIndex].\n * \n * @sample
samples.collections.Arrays.Sorting.sortRangeOfArrayOfComparable\n */\n@SinceKotlin(\"1.4\")\npublic expect
fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range
in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param
toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample
samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun
ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n *
@param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range
(exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is
less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if
[fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n
*/\n@SinceKotlin(\"1.4\")\npublic expect fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n
* Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by
default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * \n * @throws
IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * \n * @sample
samples.collections.Arrays.Sorting.sortRangeOfArray\n */\n@SinceKotlin(\"1.4\")\npublic expect fun
IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n * Sorts a range in the array in-place.\n * \n *
@param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range
(exclusive) to sort, size of this array by default.\n * \n * @throws IndexOutOfBoundsException if [fromIndex] is
less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if
[fromIndex] is greater than [toIndex].\n * \n * @sample samples.collections.Arrays.Sorting.sortRangeOfArray\n
*/\n@SinceKotlin(\"1.4\")\npublic expect fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit\n\n/**\n
* Sorts a range in the array in-place.\n * \n * @param fromIndex the start of the range (inclusive) to sort, 0 by

Open Source Used In DNAC Platform Arica 7742


@param toIndex the end of the range (exclusive) to sort, size of this array by default.
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

* @param fromIndex the start of the range (inclusive) to sort, 0 by default.

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@sample samples.collections.Arrays.Sorting.sortRangeOfArray

*/
@SinceKotlin("1.4")
public expect fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts a range in the array in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 * 
 */
@SinceKotlin("1.4")
public expect fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 * 
 */
@SinceKotlin("1.4")
public expect fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Array<out T>.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sortWith(reverseOrder(), fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
public fun ByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
public fun ShortArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
public fun IntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
public fun LongArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}
fromIndex the start of the range (inclusive) to sort.
* @param toIndex the end of the range (exclusive) to sort.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of
this array.
* @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.4")
public fun FloatArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of
this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.4")
public fun DoubleArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of
this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.4")
public fun CharArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param comparator the comparator to use.
 * @throws IllegalArgumentException if [comparator] is null.

public expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit

/**
 * Sorts a range in the array in-place with the given [comparator].
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

public expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns an array of Boolean containing all of the elements of this generic array.
 * @returns an array of Boolean containing all of the elements of this generic array.

public fun Array<out Boolean>.toBooleanArray(): BooleanArray {
    return BooleanArray(size) { index -> this[index] }
}

/**
 * Returns an array of Byte containing all of the elements of this generic array.
 * @returns an array of Byte containing all of the elements of this generic array.

public fun Array<out Byte>.toByteArray(): ByteArray {
    return ByteArray(size) { index -> this[index] }
}

/**
 * Returns an array of Char containing all of the elements of this generic array.
 * @returns an array of Char containing all of the elements of this generic array.

public fun Array<out Char>.toCharArray(): CharArray {
    return CharArray(size) { index -> this[index] }
}

/**
 * Returns an array of Double containing all of the elements of this generic array.
 * @returns an array of Double containing all of the elements of this generic array.

public fun Array<out Double>.toDoubleArray(): DoubleArray {
    return DoubleArray(size) { index -> this[index] }
}

/**
 * Returns an array of Float containing all of the elements of this generic array.
 * @returns an array of Float containing all of the elements of this generic array.

public fun Array<out Float>.toFloatArray(): FloatArray {
    return FloatArray(size) { index -> this[index] }
}

/**
 * Returns an array of Int containing all of the elements of this generic array.
 * @returns an array of Int containing all of the elements of this generic array.

public fun Array<out Int>.toIntArray(): IntArray {
    return IntArray(size) { index -> this[index] }
}

/**
 * Returns an array of Long containing all of the elements of this generic array.
 * @returns an array of Long containing all of the elements of this generic array.

public fun Array<out Long>.toLongArray(): LongArray {
    return LongArray(size) { index -> this[index] }
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 * @returns a *typed* object array containing all of the elements of this primitive array.

public expect fun ByteArray.toTypedArray(): Array<Byte>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 * @returns a *typed* object array containing all of the elements of this primitive array.

public expect fun ShortArray.toTypedArray(): Array<Short>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 * @returns a *typed* object array containing all of the elements of this primitive array.

public expect fun IntArray.toTypedArray(): Array<Int>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 * @returns a *typed* object array containing all of the elements of this primitive array.

public expect fun LongArray.toTypedArray(): Array<Long>
*typed* object array containing all of the elements of this primitive array.

```kotlin
public expect fun DoubleArray.toTypedArray(): Array<Double>
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
public expect fun CharArray.toTypedArray(): Array<Char>
```

The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map.
map).
The returned map preserves the entry iteration order of the original array.

```kotlin
BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}
```

```kotlin
CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}
```

```kotlin
Array<out T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
}
```

```kotlin
ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Byte>(capacity), keySelector)
}
```

```kotlin
ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Short>(capacity), keySelector)
}
```

```kotlin
IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Int>(capacity), keySelector)
}
```

```kotlin
LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Long>(capacity), keySelector)
}
```
order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
public inline fun <K> FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Float>(capacity),
        keySelector)/* Returns a [Map] containing the elements from the given array indexed by the key
        returned from [keySelector] function applied to each element. If any two elements would have the same
        key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
public inline fun <K> DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Double>(capacity),
        keySelector)/* Returns a [Map] containing the elements from the given array indexed by the key
        returned from [keySelector] function applied to each element. If any two elements would have the same
        key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
public inline fun <K> BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Boolean>(capacity),
        keySelector)/* Returns a [Map] containing the elements from the given array indexed by the key
        returned from [keySelector] function applied to each element. If any two elements would have the same
        key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
public inline fun <K> CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Char>(capacity),
        keySelector)/* Returns a [Map] containing the elements from the given array indexed by the key
        returned from [keySelector] function applied to each element. If any two elements would have the same
        key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform
public inline fun <T, K, V> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
        keySelector, valueTransform)/* Returns a [Map] containing the values provided by [valueTransform]
        and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have
        the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform
public inline fun <T, K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
        keySelector, valueTransform)/* Returns a [Map] containing the values provided by [valueTransform]
        and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have
        the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform
public inline fun <T, K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
        keySelector, valueTransform)/* Returns a [Map] containing the values provided by [valueTransform]
        and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have
        the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration
        order of the original array. */
}
have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> CharArray.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)\n    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}\n\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */
public inline fun <T, K, M :MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */
public inline fun <T, K, M :MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */
public inline fun <T, K, M :MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n * where key is provided by the [keySelector] function applied to each element of the given array.\n * \n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */
public inline fun <T, K, M :MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)\n    }\n    return destination\n}
[keySelector] function applied to each element of the given array
* and value is the element itself.
* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
* @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

```kotlin
@public inline fun <K, M : MutableMap<in K, in Byte>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K): M {
     for (element in this) {
         destination.put(keySelector(element), element)
     }
     return destination
}
```

Populates and returns the [destination] mutable map with key-value pairs.
* where key is provided by the [keySelector] function applied to each element of the given array
* and value is the element itself.
* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

```kotlin
@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

```
public inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array.

 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform()
 */
public inline fun <K, V, M : MutableMap<K, in V>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform()
 */
public inline fun <K, V, M : MutableMap<K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <T, K, V, M : MutableMap<K, in V>> Array<out T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> ByteArray.associateTo(destination: M, transform: (Byte) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> LongArray.associateTo(destination: M, transform: (Long) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> CharSequence.associateTo(destination: M, transform: (CharSequence) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any of two pairs would have the same key the last one gets added to the map. * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo()
 */
public inline fun <K, V, M : MutableMap<K, in V>> Any.associateTo(destination: M, transform: (Any) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each
element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */
public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) ->
Pair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}*\n/**
 * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V>
ByteArray.associateWith(valueSelector: (Byte) -> V): Map<Byte, V> {
    val result = LinkedHashMap<Byte,
V>(mapCapacity(size).coerceAtLeast(16))\n    return associateWithTo(result,
valueSelector)\n}*\n/**
 * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V>
ShortArray.associateWith(valueSelector: (Short) -> V): Map<Short, V> {
    val result = LinkedHashMap<Short,
V>(mapCapacity(size).coerceAtLeast(16))\n    return associateWithTo(result,
valueSelector)\n}*\n/**
 * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * \n * If any of two pairs would have the same key the last one gets added to the
map.\n * \n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo\n */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V>
IntArray.associateWith(valueSelector: (Int) -> V): Map<Int, V> {
    val result = LinkedHashMap<Int,
V>(mapCapacity(size).coerceAtLeast(16))\n    return associateWithTo(result,
valueSelector)\n}*\n
LongArray.associateWith(valueSelector: (Long) -> V): Map<Long, V> {
    val result = LinkedHashMap<Long, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
} *\n
FloaArray.associateWith(valueSelector: (Float) -> V): Map<Float, V> {
    val result = LinkedHashMap<Float, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
} *\n
DoubleArray.associateWith(valueSelector: (Double) -> V): Map<Double, V> {
    val result = LinkedHashMap<Double, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
} *\n
CharArray.associateWith(valueSelector: (Char) -> V): Map<Char, V> {
    val result = LinkedHashMap<Char, V>(mapCapacity(size.coerceAtMost(128)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
} *\n
Array<out K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
} *\n
ByteArray.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
} *\n
CharArray.associateWithTo(destination: M, valueSelector: (Char) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
} *\n
ByteArray.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
} *\n
ByteBuffer.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
} *\n
Open Source Used In DNAC Platform Arica 7753
ShortArray.associateWithTo(destination: M, valueSelector: (Short) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

IntArray.associateWithTo(destination: M, valueSelector: (Int) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

LongArray.associateWithTo(destination: M, valueSelector: (Long) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

FloatArray.associateWithTo(destination: M, valueSelector: (Float) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

DoubleArray.associateWithTo(destination: M, valueSelector: (Double) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

CharArray.associateWithTo(destination: M, valueSelector: (Char) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}  

// mutable map with key-value pairs for each element of the given array,
* where key is the element itself and value is provided by the [valueSelector] function applied to that key.

* If any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

Array.associateWithTo(destination: M, valueSelector: (T) -> V): M {
    for (item in this) {
        destination.add(item)
    }
    return destination
}  

// Appends all elements to the given [destination] collection.
* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.

samples.collections.Collections.Transformations.associateWithTo

* Append all elements to the given [destination] collection.

* if any two elements are equal, the last one overwrites the former value in the map.
destination.add(item)  

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Int>> IntArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Long>> LongArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Double>> DoubleArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <C : MutableCollection<in Boolean>> BooleanArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun <T> Array<out T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun ByteArray.toHashSet(): HashSet<Byte> {
    return toCollection(HashSet<Byte>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun ShortArray.toHashSet(): HashSet<Short> {
    return toCollection(HashSet<Short>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun IntArray.toHashSet(): HashSet<Int> {
    return toCollection(HashSet<Int>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun LongArray.toHashSet(): HashSet<Long> {
    return toCollection(HashSet<Long>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun FloatArray.toHashSet(): HashSet<Float> {
    return toCollection(HashSet<Float>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun DoubleArray.toHashSet(): HashSet<Double> {
    return toCollection(HashSet<Double>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun BooleanArray.toHashSet(): HashSet<Boolean> {
    return toCollection(HashSet<Boolean>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun CharArray.toHashSet(): HashSet.CharArray(toCollection(HashSet<Char>(mapCapacity(size.coerceAtMost(128)))){
    return toCollection(HashSet<Char>(mapCapacity(size.coerceAtMost(128))))
})

/**
 * Returns a [List] containing all elements.
 */
public fun <T> Array<out T>.toList(): List<T> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun ByteArray.toList(): List<Byte> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun ShortArray.toList(): List<Short> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun IntArray.toList(): List<Int> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun LongArray.toList(): List<Long> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun FloatArray.toList(): List<Float> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun DoubleArray.toList(): List<Double> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}
listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun BooleanArray.toList(): List<Boolean> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun CharArray.toList(): List<Char> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    val list = ArrayList<Byte>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ShortArray.toMutableList(): MutableList<Short> {
    val list = ArrayList<Short>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun IntArray.toMutableList(): MutableList<Int> {
    val list = ArrayList<Int>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun LongArray.toMutableList(): MutableList<Long> {
    val list = ArrayList<Long>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun FloatArray.toMutableList(): MutableList<Float> {
    val list = ArrayList<Float>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun DoubleArray.toMutableList(): MutableList<Double> {
    val list = ArrayList<Double>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun BooleanArray.toMutableList(): MutableList<Boolean> {
    val list = ArrayList<Boolean>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun CharArray.toMutableList(): MutableList<Char> {
    val list = ArrayList<Char>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ShortArray.toMutableList(): MutableList<Short> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun IntArray.toMutableList(): MutableList<Int> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun LongArray.toMutableList(): MutableList<Long> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun FloatArray.toMutableList(): MutableList<Float> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun DoubleArray.toMutableList(): MutableList<Double> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun BooleanArray.toMutableList(): MutableList<Boolean> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun CharArray.toMutableList(): MutableList<Char> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun ByteArray.toSet(): Set<Byte> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Byte>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun ShortArray.toSet(): Set<Short> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Short>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun IntArray.toSet(): Set<Int> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Int>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun LongArray.toSet(): Set<Long> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Long>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun FloatArray.toSet(): Set<Float> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Float>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun BooleanArray.toSet(): Set<Boolean> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
 */
public fun CharArray.toSet(): Set<Char> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Char>(mapCapacity(size)))
    }
}
elements.

* The returned set preserves the element iteration order of the original array.

```kotlin
@sample
fun CharArray.toSet(): Set<Char> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet(Char(mapCapacity(size.coerceAtMost(128)))))
    }
}
```

```kotlin
/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```kotlin
/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
```
@sample samples.collections.Collections.Transformations.flatMapIndexed

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterable")\n@kotlin.internal.InlineOnly
\npublic inline fun <R> ByteArray.flatMapIndexed(transform: (index: Int, Byte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> ShortArray.flatMapIndexed(transform: (index: Int, Short) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> IntArray.flatMapIndexed(transform: (index: Int, Int) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> LongArray.flatMapIndexed(transform: (index: Int, Long) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> FloatArray.flatMapIndexed(transform: (index: Int, Float) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> DoubleArray.flatMapIndexed(transform: (index: Int, Double) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> BooleanArray.flatMapIndexed(transform: (index: Int, Boolean) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from
 * results of [transform] function being invoked on each element * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
\npublic inline fun <R> CharArray.flatMapIndexed(transform: (index: Int, Char) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
@sample samples.collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.Transformations.flatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed

@sample samples.collections.Collections.TransformationsflatMapIndexed
public inline fun <R, C : MutableCollection<in R>> DoubleArray.flatMapIndexedTo(destination: C, transform: (index: Int, Double) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element * and its index in the original array, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> BooleanArray.flatMapIndexedTo(destination: C, transform: (index: Int, Boolean) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element * and its index in the original array, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> CharArray.flatMapIndexedTo(destination: C, transform: (index: Int, Char) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element) 
        destination.addAll(list) 
    }
    return destination 
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element) 
        destination.addAll(list) 
    }
    return destination 
}
list = transform(element) |n destination.addAll(list) |n return destination

// Append all elements yielded from results of [transform] function being invoked on each element of original array, to the given
// destination.

@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
public inline fun <R, C : MutableCollection<R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequenceTo")
public inline fun <T, R, C : MutableCollection<R>> Array<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
 * corresponding elements.
 */
public inline fun <T, K> Array<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

// Append all elements yielded from results of [transform] function being invoked on each
// element of original array, to the given [destination].

entry iteration order of the keys produced from the original array.

```kotlin
public inline fun <K, L, V> Iterable<out T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

The returned map preserves the entry iteration order of the keys produced from the original array.

```kotlin
public inline fun <K, V> ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

The returned map preserves the entry iteration order of the keys produced from the original array.
returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n *\npublic inline fun <K, V>
LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {
    return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform).\n}

Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n *\npublic inline fun <K, V>
FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {
    return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform).\n}

Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n *\npublic inline fun <K, V>
DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {
    return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform).\n}

Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n *\npublic inline fun <K, V>
BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>> {
    return
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform).\n}

Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues

Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n
Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * \n * @return The [destination] map.\n * \n * @sample samples.collections.Collections.Transformations.groupBy

Open Source Used In DNAC Platform Arica

7763
[destination] map each group key associated with a list of corresponding elements.

```
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<?>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Short>() } list.add(element) } return destination
}

```

```
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<?>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Int>() } list.add(element) } return destination
}

```

```
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<?>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Long>() } list.add(element) } return destination
}

```

```
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M :MutableMap<in K, MutableList<?>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Float>() } list.add(element) } return destination
}

```

```
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<?>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Double>() } list.add(element) } return destination
}

```

```
/**
 * Groups values returned by the [valueTransform] function
 * applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<?>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() } list.add(element) } return destination
}

```
destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableMap<K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableList<MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableList<MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableList<MutableList<V>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableList<MutableList<V>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() } \n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/

public inline fun <K, V, M : MutableList<MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {\n for (element in this) {\n val key = keySelector(element)\n val list = destination.getOrPut(key) { ArrayList<V>() }\n list.add(valueTransform(element))\n }\n return
destination\n\n**Groups values returned by the [valueTransform] function applied to each element of the original array\n** by the key returned by the given [keySelector] function applied to the element\nand puts to the [destination] map each group key associated with a list of corresponding values.\n\n\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n*/


MutableMap<in K, MutableList<V>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group associated with a list of corresponding values.
 * @return The [destination] map.
 */
public inline fun <K, V, M :.MutableMap<in K, MutableList<V>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * @return A [Grouping] source.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <T, R> Array<out T>.map(transform: (T) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> ByteArray.map(transform: (Byte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> ShortArray.map(transform: (Short) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> IntArray.map(transform: (Int) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> LongArray.map(transform: (Long) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> FloatArray.map(transform: (Float) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
public inline fun <R> DoubleArray.map(transform: (Double) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
CharArray.map(transform: (Char) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 * @return A list containing the results of applying the given [transform] function.
 */
CharArray.mapIndexed(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
element and its index in the original array.

* @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

*public inline fun <R> ByteArray.mapIndexed(transform: (index: Int, Byte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> ShortArray.mapIndexed(transform: (index: Int, Short) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> IntArray.mapIndexed(transform: (index: Int, Int) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> LongArray.mapIndexed(transform: (index: Int, Long) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> FloatArray.mapIndexed(transform: (index: Int, Float) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> DoubleArray.mapIndexed(transform: (index: Int, Double) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> BooleanArray.mapIndexed(transform: (index: Int, Boolean) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <R> CharArray.mapIndexed(transform: (index: Int, Char) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.

*public inline fun <T> FloatArray.mapIndexedNotNull(transform: (index: Int, Float) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> ShortArray.mapIndexedNotNull(transform: (index: Int, Short) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> IntArray.mapIndexedNotNull(transform: (index: Int, Int) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> LongArray.mapIndexedNotNull(transform: (index: Int, Long) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> FloatArray.mapIndexedNotNull(transform: (index: Int, Float) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> DoubleArray.mapIndexedNotNull(transform: (index: Int, Double) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> BooleanArray.mapIndexedNotNull(transform: (index: Int, Boolean) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T> CharArray.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform) * Applies the given [transform] function to each element and its index in the original array and returns the result of the transform applied to the element.

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T, R) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination

*public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
[transform] function to each element and its index in the original array\n * and appends the results to the given
 [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */\n@sample samples.collections.Collections.Transformations.mapIndexed

public inline fun <S, T, C : MutableCollection<in S>> S.mapIndexedTo(destination: C, transform: (index: Int, S) -> T): C {
    var index = 0
    for (item in this)
        if (item != null)
            destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given
 [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@sample samples.collections.Collections.Transformations.mapNotNull

public inline fun <S, T, C : Collection<S>> S.mapIndexedTo(destination: C, transform: (index: Int, S) -> T?): C {
    var index = 0
    for (item in this)
        if (item != null)
            destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array\n * and appends the results to the given
 [destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@sample samples.collections.Collections.Transformations.map

public inline fun <S, T, C : Collection<S>> S.mapIndexedTo(destination: C, transform: (index: Int, S) -> T): C {
    var index = 0
    for (item in this)
        if (item != null)
            destination.add(transform(index++, item))
    return destination
}
```kotlin
// An public inline fun <T, R, C : MutableCollection<R>> Array<out T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array and
 * and appends the results to the given [destination].
 */
fun <R, C : MutableCollection<R>> byteArray.mapTo(destination: C, transform: (Byte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C : MutableCollection<R>> shortArray.mapTo(destination: C, transform: (Short) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C : MutableCollection<R>> intArray.mapTo(destination: C, transform: (Int) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C : MutableCollection<R>> longArray.mapTo(destination: C, transform: (Long) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C : MutableCollection<R>> floatArray.mapTo(destination: C, transform: (Float) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C :MutableCollection<R>> doubleArray.mapTo(destination: C, transform: (Double) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C :MutableCollection<R>> booleanArray.mapTo(destination: C, transform: (Boolean) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
fun <R, C :MutableCollection<R>> charArray.mapTo(destination: C, transform: (Char) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array into an
 * [IndexedValue] containing the index of that element and the element itself.
 */
fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }
}
```
Returns a lazy [Iterable] that wraps each element of the original array into an [IndexedValue] containing the index of that element and the element itself.

```kotlin
public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun ByteArray.distinct(): List<Byte> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun ShortArray.distinct(): List<Short> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun IntArray.distinct(): List<Int> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun LongArray.distinct(): List<Long> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only distinct elements from the given array.

```kotlin
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList()
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <K> ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {
    val set = HashSet<K>()
    val list = ArrayList<Byte>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```

Returns a list containing only elements from the given array having distinct keys returned by the given [selector] function.

```kotlin
public inline fun <T, K> Array<out T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
```
were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

ShortArray.distinctBy(selector: (Short) -> K): List<Short> {
    val set = HashSet<K>()
    val list = ArrayList<Short>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

IntArray.distinctBy(selector: (Int) -> K): List<Int> {
    val set = HashSet<K>()
    val list = ArrayList<Int>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

LongArray.distinctBy(selector: (Long) -> K): List<Long> {
    val set = HashSet<K>()
    val list = ArrayList<Long>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

FloatArray.distinctBy(selector: (Float) -> K): List<Float> {
    val set = HashSet<K>()
    val list = ArrayList<Float>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

DoubleArray.distinctBy(selector: (Double) -> K): List<Double> {
    val set = HashSet<K>()
    val list = ArrayList<Double>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

CharArray.distinctBy(selector: (Char) -> K): List<Char> {
    val set = HashSet<K>()
    val list = ArrayList<Char>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key)) list.add(e)
    }
    return list
}

* The elements in the resulting list are in the same order as they were in the source array.

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <K>

CharArray.intersect(other: Iterable<Char>): Set<Char> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <T>

Array<out T>.intersect(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* @sample

samples.collections.Collections.Transformations.distinctAndDistinctBy

*\n\npublic inline fun <Byte>

ByteArray.intersect(other: Iterable<Byte>): Set<Byte> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* Returns a set containing all elements that are contained by both this array and the specified collection.

* @sample

Open Source Used In DNAC Platform Arica  7771
The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained at least in one of these collections.

```
val set = this.toMutableSet()
set.removeAll(other)
return set
```

* Returns a set containing all elements that are contained by this array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* The returned set preserves the element iteration order of the original array.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```

* Returns a set containing all elements that are contained by both this array and the specified collection.

```
val set = this.toMutableSet()
set.retainAll(other)
return set
```
array and not contained by the specified collection. The returned set preserves the element iteration order of the original array. *

public infix fun DoubleArray.subtract(other: Iterable<Double>): Set<Double> { val set = this.toMutableSet() set.removeAll(other) return set }

public infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Byte> { val set = this.toMutableSet() set.removeAll(other) return set }
that are contained in both collections use [intersect].\n */
public infix fun ShortArray.union(other: Iterable<Int>): Set<Int> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n} \n
/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the
 * order of the [other] collection.\n */
public infix fun IntArray.union(other: Iterable<Long>): Set<Long> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n}

/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the
 * order of the [other] collection.\n */
public infix fun LongArray.union(other: Iterable<Float>): Set<Float> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n}

/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the
 * order of the [other] collection.\n */
public infix fun FloatArray.union(other: Iterable<Double>): Set<Double> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n}

/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the
 * order of the [other] collection.\n */
public infix fun DoubleArray.union(other: Iterable<Boolean>): Set<Boolean> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n}

/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the
 * order of the [other] collection.\n */
public infix fun BooleanArray.union(other: Iterable<Char>): Set<Char> {\n    \n    val set = this.toMutableSet()\n    set.addAll(other)\n    return set\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun <T> Array<out T>.all(predicate: (T) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun ByteArray.all(predicate: (Byte) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all\n */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {\n    for (element in this) if (!predicate(element)) return false\n    return true\n}
DoubleArray.all(predicate: (Double) -> Boolean): Boolean { for (element in this) if (!predicate(element)) return false
    return true // Returns 'true' if all elements match the given [predicate].
}

doubleArray.all() { // Returns 'true' if array has at least one element.
    return any { // Returns 'true' if at least one element.
        a in this
        a.equals(2.0)
    }
}

@sample
Collections.Aggregates.allUn(samples.collections.Collections.Aggregates.allUn(*@public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean { for (element in this) if (!predicate(element)) return false
    return true // Returns 'true' if all elements match the given [predicate].
}

public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean { for (element in this) if (!predicate(element)) return false
    return true // Returns 'true' if all elements match the given [predicate].
}

public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean { for (element in this) if (!predicate(element)) return false
    return true // Returns 'true' if all elements match the given [predicate].
}

public fun <T> Array<out T>.any(): Boolean { return !isEmpty()
}

public fun ByteArray.any(): Boolean { return !isEmpty()
}

public fun ShortArray.any(): Boolean { return !isEmpty()
}

public fun IntArray.any(): Boolean { return !isEmpty()
}

public fun LongArray.any(): Boolean { return !isEmpty()
}

public fun FloatArray.any(): Boolean { return !isEmpty()
}

public fun DoubleArray.any(): Boolean { return !isEmpty()
}

public fun BooleanArray.any(): Boolean { return !isEmpty()
}

public fun CharArray.any(): Boolean { return !isEmpty()
}

public fun AnyArray.anyWithPredicate(predicate: (T) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun ByteArray.anyWithPredicate(predicate: (Byte) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun ShortArray.anyWithPredicate(predicate: (Short) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun IntArray.anyWithPredicate(predicate: (Int) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun LongArray.anyWithPredicate(predicate: (Long) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun FloatArray.anyWithPredicate(predicate: (Float) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun DoubleArray.anyWithPredicate(predicate: (Double) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}

public fun BooleanArray.anyWithPredicate(predicate: (Boolean) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true
    return false
}
false

Returns `true` if at least one element matches the given [predicate].

```kotlin
fun <T> count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun ByteArray.count(predicate: (Byte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun ShortArray.count(predicate: (Short) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun IntArray.count(predicate: (Int) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```

```kotlin
fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
```
current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> IntArray.fold(initial: R, operation: (acc: R, Int) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n */\npublic inline fun <T> ByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
} // * Accumulates value starting with [initial] value and applying [operation] from left to right\n * to current accumulator value and each element.\n * Returns the specified [initial] value if the array is empty.\n * @param [operation] function that takes the
index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n */
public inline fun <R> ShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right.\n * Returns the specified [initial] value if the array is empty.\n */
public inline fun <T, R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left.\n * Returns the specified [initial] value if the array is empty.\n */
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left.\n * Returns the specified [initial] value if the array is empty.\n */
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
The public inline function `Array.foldRight` is defined as follows:

```kotlin
public inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

It takes an initial value and an operation function as parameters. The operation function takes the index of an element, the element itself, and the current accumulator value, and returns the next accumulator value. If the array is empty, it returns the specified initial value.

For example, consider the following code snippet:

```kotlin
ShortArray.foldRight(initial: R, operation: (Short, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

This code accumulates value starting with the initial value and applying the operation function from right to left for each element and current accumulator value, and returns the specified initial value if the array is empty.
operation(index, get(index), accumulator)\n --operator--\n \n return accumulator
\n
/*\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> ShortArray.foldRightIndexed(initial: R, operation: (index: Int, Short, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> IntArray.foldRightIndexed(initial: R, operation: (index: Int, Int, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> LongArray.foldRightIndexed(initial: R, operation: (index: Int, Long, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> ByteArray.foldRightIndexed(initial: R, operation: (index: Int, Byte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}

/**\nAccumulates value starting with [initial] value and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n\n*\n*\n*\n* Returns the specified [initial] value if the array is empty.\n*\n*\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> FloatArray.foldRightIndexed(initial: R, operation: (index: Int, Float, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initia1\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n        --index\n    }
    return accumulator
}
@param \[operation\] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```kotlin
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
/**
 * Performs the given \[action\] on each element.*
 */
public inline fun ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun BooleanArray.forEachIndexed(action: (index: Int, Boolean) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
/**
 * Performs the given \[action\] on each element, providing sequential index with the element.*
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    for (element in this) action(element)
}
```
the action on the element.

```kotlin
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Array<out Double>.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.max(): Byte? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.max(): Short? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.max(): Int? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.max(): Long? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun FloatArray.max(): Float? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun DoubleArray.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharArray.max(): Char? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.max(): Byte? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.max(): Short? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.max(): Int? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.max(): Long? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun FloatArray.max(): Float? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun DoubleArray.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharArray.max(): Char? {
    return maxOrNull()
}
```

Open Source Used In DNAC Platform Arica 7782
maxByOrNull(selector): Int? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): Byte? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): Short? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): Long? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): Double? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): Float? * Returns the first element yielding the largest value of the given function
    or `null` if there are no elements.

maxByOrNull(selector): T? * Returns the first element yielding the largest value of the given function or `null` if there are no elements.

*\@SinceKotlin("1.4"))*\npublic inline fun <T, R : Comparable<R>> Array<out T>.maxByOrNull(selector: (T) -> R): T? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> Array<Byte>.maxByOrNull(selector: (Byte) -> R): Byte? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> IntArray.maxByOrNull(selector: (Int) -> R): Int? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> LongArray.maxByOrNull(selector: (Long) -> R): Long? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> FloatArray.maxByOrNull(selector: (Float) -> R): Float? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> DoubleArray.maxByOrNull(selector: (Double) -> R): Double? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> ByteArray.maxByOrNull(selector: (Byte) -> R): Byte? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}

*\@SinceKotlin("1.4"))*\npublic inline fun <R : Comparable<R>> BooleanArray.maxByOrNull(selector: (Boolean) -> R): Boolean? {\n    if (isEmpty()) return null\n    var maxElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return maxElem\n    var maxValue = selector(maxElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (maxValue < v) {\n            maxElem = e\n            maxValue = v\n        }\n    }\n    return maxElem\n}
(i in 1..lastIndex) { \n    val v = selector(this[i]) \n    maxValue = maxOf(maxValue, v) \n} \nreturn \nmaxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOf(selector: (Double) -> Double) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOf(selector: (Boolean) -> Boolean) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.maxOf(selector: (Char) -> Char) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.maxOf(selector: (T) -> Float) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.maxOf(selector: (Byte) -> Float) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOf(selector: (Short) -> Float) {\n    if (isEmpty()) throw NoSuchElementException() \n    var maxValue = selector(this[0]) \n    for (i in 1..lastIndex) { \n        val v = selector(this[i]) \n        maxValue = maxOf(maxValue, v) \n    } \n    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.\n * \n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOf(selector: (Int) -> Float) {
Float {\n    if (isInstantiationException) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n *\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@public inline fun FloatArray.maxOf(selector: (Float) -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n *\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@public inline fun DoubleArray.maxOf(selector: (Double) -> Double): Double {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n *\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n *\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@public inline fun CharArray.maxOf(selector: (Char) -> Float): Float {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n *\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@public inline fun <T, R : Comparable<R>> Array<out T>.maxOf(selector: (T) -> R): R {\n    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
Open Source Used In DNAC Platform Arica  7786
ByteArray.maxOf(selector: (Byte) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
ShortArray.maxOf(selector: (Short) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
IntArray.maxOf(selector: (Int) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
LongArray.maxOf(selector: (Long) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
FloatArray.maxOf(selector: (Float) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
DoubleArray.maxOf(selector: (Double) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
BooleanArray.maxOf(selector: (Boolean) -> R): R

    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
CharArray.maxOf(selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ByteArray.maxOfOrNull(selector: (Byte) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ShortArray.maxOfOrNull(selector: (Short) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.maxOfOrNull(selector: (Int) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun LongArray.maxOfOrNull(selector: (Long) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByNameLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun FloatArray.maxOfOrNull(selector: (Float) -> Double): Double? {
    if (isEmpty()) return null
    var
    varValue = selector(this[0])
    for (i in 1..lastIndex) {
        var
        varValue = selector(this[i])
        if (maxValue < v) {
            maxValue = maxOf(maxValue, v)
        }
    }
    return maxValue
}
public inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun CharArray.maxOfOrNull(selector: (Char) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun ByteArray.maxOfOrNull(selector: (Byte) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun ShortArray.maxOfOrNull(selector: (Short) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * * * * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

public inline fun IntArray.maxOfOrNull(selector: (Int) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```kotlin
/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun LongArray.maxOfOrNull(selector: (Long) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun FloatArray.maxOfOrNull(selector: (Float) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharArray.maxOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is
 * `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun ByteArray.maxOfOrNull(selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

ByLambdaReturnType\n@kotlin.internal.InlineOnly\n\n\npublic inline fun <R : Comparable<R>>
ShortArray.maxOfOrNull(selector: (Short) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
IntArray.maxOfOrNull(selector: (Int) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
LongArray.maxOfOrNull(selector: (Long) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
FloatArray.maxOfOrNull(selector: (Float) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
DoubleArray.maxOfOrNull(selector: (Double) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
BooleanArray.maxOfOrNull(selector: (Boolean) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced
 * by [selector] function * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
CharArray.maxOfOrNull(selector: (Char) -> R): R? {\n    if (isEmpty()) return null\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (maxValue < v) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value according to the provided
 * [comparator] among all values produced by [selector] function applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value according to the provided [comparator] among all values
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        if (comparator.compare(maxValue, v) < 0) {\n            maxValue = v\n        }\n    }\n    return maxValue\n}
produced by [selector] function applied to each element in the array.

```kotlin
*@

*@

*@

*@

*@

*@

*@

*@

*@

*@

*@

*@

*@

*@

*@
```
(comparator.compare(maxValue, ν) < 0) {\n    maxValue = ν \n    }\n\nreturn maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> CharArray.maxOfWith(\n        comparator: Comparator<in R>, selector: (Char) -> R): R {\n    if (isNullOrEmpty()) throw NoSuchElementException();\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <T, R> Array<out T>.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (T) -> R): R? {\n    if (isEmpty()) return null;\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (Byte) -> R): R? {\n    if (isEmpty()) return null;\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (Short) -> R): R? {\n    if (isEmpty()) return null;\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (Int) -> R): R? {\n    if (isEmpty()) return null;\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> LongArray.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (Long) -> R): R? {\n    if (isEmpty()) return null;\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType@kotlin.internal.InlineOnly\npublic inline fun <R> FloatArray.maxOfWithOrNull(\n        comparator: Comparator<in R>, selector: (Float) -> R): R? {\n    if (isNullOrEmpty()) throw NoSuchElementException();\n    var maxValue = selector(this[0]);\n    for (i in 1..lastIndex) {\n        var val = selector(this[i]);\n        if (comparator.compare(maxValue, val) < 0) {\n            maxValue = val;\n        }\n    }\n    return maxValue;\n}/**\n * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.\n */
return null
var maxValue = selector(this[0])
for (i in 1..lastIndex)
    if (comparator.compare(maxValue, v) < 0)
        maxValue = v

return maxValuen

Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly

public inline fun <R>
DoubleArray.maxOfWithOrNull(comparator: Comparator<R>, selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex)
        if (comparator.compare(maxValue, v) < 0)
            maxValue = v
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or ‘null’ if there are no elements.
 * 
 * If any of elements is NaN returns NaN.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly

public inline fun <R>
BooleanArray.maxOfWithOrNull(comparator: Comparator<R>, selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex)
        if (comparator.compare(maxValue, v) < 0)
            maxValue = v
    return maxValue
}

/**
 * Returns the largest element or ‘null’ if there are no elements.
 * 
 * If any of elements is NaN returns NaN.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly

public inline fun <T : Comparable<T>> Array<out T>.maxOrNull(): T? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ByteArray.maxOrNull(): Byte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ShortArray.maxOrNull(): Short? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun IntArray.maxOrNull(): Int? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun LongArray.maxOrNull(): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is NaN returns NaN.
 */
@SinceKotlin("1.4")
public fun FloatArray.maxOrNull(): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is NaN returns NaN.
 */
@SinceKotlin("1.4")
public fun DoubleArray.maxOrNull(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e)
            max = e
    return max
}
return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] max = maxOf(max, e) } return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.maxOrNull(): Char? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (max < e) max = e } return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun <T> Array<out T>.maxWith(comparator: Comparator<in T>): T? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.maxWithOrNull(comparator: Comparator<in Char>): Char? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ByteArray.maxWithOrNull(comparator: Comparator<in Byte>): Byte? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ShortArray.maxWithOrNull(comparator: Comparator<in Short>): Short? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun IntArray.maxWithOrNull(comparator: Comparator<in Int>): Int? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun LongArray.maxWithOrNull(comparator: Comparator<in Long>): Long? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun FloatArray.maxWithOrNull(comparator: Comparator<in Float>): Float? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun DoubleArray.maxWithOrNull(comparator: Comparator<in Double>): Double? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun BooleanArray.maxWithOrNull(comparator: Comparator<in Boolean>): Boolean? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.maxWithOrNull(comparator: Comparator<in Char>): Char? { if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(max, e) < 0) max = e } return max
}
```kotlin
@SinceKotlin("1.4")
public fun LongArray.maxWithOrNull(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun FloatArray.maxWithOrNull(comparator: Comparator<in Float>): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun DoubleArray.maxWithOrNull(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun BooleanArray.maxWithOrNull(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Array<out Double>.min(): Double? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Array<out Float>.min(): Float? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Array<out T>.min(): T? {
    return minByOrNull()
}

@Deprecated("Use minByOrNull instead.",
ReplaceWith("minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {
    return minByOrNull(selector)
}
```

---

Open Source Used In DNAC Platform Arica  7796
ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> DoubleArray.minBy(selector: (Double) -> R): Double? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

ReplaceWith("minByOrOrNull(selector)")

@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharArray.minBy(selector: (Char) -> R): Char? {
    return minByOrOrNull(selector) @n@ @Deprecated("Use minByOrNull instead.",
}

/**
* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
* @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")
public inline fun <T, R : Comparable<R>> Array<out T>.minByOrNull(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem) for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
* @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> ByteArray.minByOrNull(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem) for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
* @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> ShortArray.minByOrNull(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem) for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
* @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> IntArray.minByOrNull(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem) for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
* @sample samples.collections.Collections.Aggregates.minByOrNull
*/
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> LongArray.minByOrNull(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem) for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}
samples.collections.Collections.Aggregates.minByOrNullOrNull */@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> FloatArray.minByOrNull(selector: (Float) -> R): Float? {
    var minElem = this[0].val
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i].val
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun <R : Comparable<R>> DoubleArray.minByOrNull(selector: (Double) -> R): Double? {
    var minElem = this[0].val
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i].val
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun <R : Comparable<R>> BooleanArray.minByOrNull(selector: (Boolean) -> R): Boolean? {
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun <R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char? {
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the smallest value among all values produced by 
 * the given function applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun <T> Array<out T>.minOf(selector: (T) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by 
 * the given function applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun ByteArray.minOf(selector: (Byte) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by 
 * the given function applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun ShortArray.minOf(selector: (Short) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by 
 * the given function applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun IntArray.minOf(selector: (Int) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by 
 * the given function applied to each element in the array.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
@public inline fun LongArray.minOf(selector: (Long) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun ByteArray.minOf(selector: (Byte) -> Float): Float {
    if (isEmpty()) throwNoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun ShortArray.minOf(selector: (Short) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun IntArray.minOf(selector: (Int) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun LongArray.minOf(selector: (Long) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun FloatArray.minOf(selector: (Float) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun DoubleArray.minOf(selector: (Double) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal:inline fun BooleanArray.minOf(selector: (Boolean) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return
}
`NaN`.

```kotlin
@NonNull
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun CharArray.minOf(selector: (Char) -> Float):
    Float {
        if (isEmpty()) throw NoSuchElementException()
        var minValue = selector(this[0])
        for (i in 1..lastIndex) {
            val v = selector(this[i])
            if (minValue > v) {
                minValue = v
            }
        }
        return minValue
    }

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Array<T>.minOf(selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ByteArray.minOf(selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ShortArray.minOf(selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> IntArray.minOf(selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> LongArray.minOf(selector: (Long) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> FloatArray.minOf(selector: (Float) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

Open Source Used In DNAC Platform Arica  7801
NoSuchElementException if the array is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>

DoubleArray.minOf(selector: (Double) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
BooleanArray.minOf(selector: (Boolean) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```
if there are no elements.\n
* If any of values produced by [selector] function is ‘NaN’, the returned result is ‘NaN’.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly
\npublic inline fun LongArray.minOfOrNull(selector: (Long) -> Double): Double? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly
\npublic inline fun FloatArray.minOfOrNull(selector: (Float) -> Double): Double? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly
\npublic inline fun DoubleArray.minOfOrNull(selector: (Double) -> Double): Double? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly
\npublic inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Double): Double? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n\npublic inline fun CharArray.minOfOrNull(selector: (Char) -> Double): Double? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.internal.InlineOnly
\npublic inline fun <T> Array<out T>.minOfOrNull(selector: (T) -> Float): Float? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n\npublic inline fun ByteArray.minOfOrNull(selector: (Byte) -> Float): Float? {\n    if (isEmpty()) return null\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue\n}
smallest value among all values produced by \([\text{selector}]\) function\n \* applied to each element in the array or `null` if there are no elements.\n \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ShortArray.minOfOrNull(selector: (Short) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.minOfOrNull(selector: (Int) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun LongArray.minOfOrNull(selector: (Long) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun FloatArray.minOfOrNull(selector: (Float) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun DoubleArray.minOfOrNull(selector: (Double) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by \([\text{selector}]\) function\n * applied to each element in the array or `null` if there are no elements.\n * \* \n * If any of values produced by \([\text{selector}]\) function is `NaN`, the returned result is `NaN`.\n
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun CharArray.minOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
val v = selector(this[i])
minValue = minOf(minValue, v)
return minValue

* Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

* @SinceKotlin("1.4")
* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)
* @OverloadResolution
* ByLambdaReturnType
* @kotlin.internal.InlineOnly
* public fun <T, R : Comparable<R>> Array<T>.minOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ByteArray.minOfOrNull(selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ShortArray.minOfOrNull(selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable>R
IntArray.minOfOrNull(selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> LongArray.minOfOrNull(selector: (Long) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> FloatArray.minOfOrNull(selector: (Float) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> DoubleArray.minOfOrNull(selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> BooleanArray.minOfOrNull(selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var minValue =
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>>
CharArray.minOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.minOfWith(comparator:
Comparator<in R>, selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.minOfWith(comparator:
Comparator<in R>, selector: (Long) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.minOfWith(comparator:
Comparator<in R>, selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWith(comparator:
Comparator<in R>, selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.minOfWith(comparator:
Comparator<in R>, selector: (Float) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
Comparator<R>, selector: (Float) -> R: R { var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.minOfWith(comparator: Comparator<R>, selector: (Double) -> R): R {
...
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.minOfWith(comparator: Comparator<R>, selector: (Boolean) -> R): R {
...
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.minOfWith(comparator: Comparator<R>, selector: (Char) -> R): R {
...
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.minOfWithOrNull(comparator: Comparator<R>, selector: (Byte) -> R): R? {
...
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWithOrNull(comparator: Comparator<R>, selector: (Short) -> R): R? {
...
}
ByLambdaReturnType

IntArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Int) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

LongArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Long) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

FloatArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Float) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

DoubleArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Double) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

BooleanArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Boolean) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

CharArray[minOfWithOrNull](comparator: Comparator<in R>, selector: (Char) -> R): R? { return null }

var minValue = selector(this[0])

for (i in 1..lastIndex) { val v = selector(this[i])
    if (comparator.compare(minValue, v) > 0) { minValue = v

    return minValue

    * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

Array[minOrNull](): T? { return null }

val e = this[i]

if (min > e) min = e

return min
return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun ByteArray.minOrNull(): Byte? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] if (min > e) min = e\n    return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun ShortArray.minOrNull(): Short? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] if (min > e) min = e\n    return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun IntArray.minOrNull(): Int? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] if (min > e) min = e\n    return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun LongArray.minOrNull(): Long? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] if (min > e) min = e\n    return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun FloatArray.minOrNull(): Float? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] min = minOf(min, e)\n    return min\n\n/**\n * Returns the smallest element or 'null' if there are no elements.\n */\n@SinceKotlin("1.4")\npublic fun DoubleArray.minOrNull(): Double? {\n    var min = this[0] for (i in 1..lastIndex) \n    val e = this[i] min = minOf(min, e)\n    return min\n\n/**\n * Returns the first element having the smallest value according to the
provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun <T> Array<out T>.minWithOrNull(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun ByteArray.minWithOrNull(comparator: Comparator<Byte>): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun ShortArray.minWithOrNull(comparator: Comparator<Short>): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun IntArray.minWithOrNull(comparator: Comparator<Int>): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun LongArray.minWithOrNull(comparator: Comparator<Long>): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun FloatArray.minWithOrNull(comparator: Comparator<Float>): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun DoubleArray.minWithOrNull(comparator: Comparator<Double>): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun BooleanArray.minWithOrNull(comparator: Comparator<Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun CharArray.minWithOrNull(comparator: Comparator<Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun <T> Array<out T>.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun ByteArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun ShortArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun IntArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun LongArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun FloatArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun DoubleArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun BooleanArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.

```kotlin
@SinceKotlin("1.4")
public fun CharArray.none(): Boolean {
    return isEmpty()
}
```

Returns `true` if the array has no elements.
isEmpty() Returns `true` if the array has no elements.

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample
 * samples.collections.Collections.Aggregates.none
 */
public fun BooleanArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample
 * samples.collections.Collections.Aggregates.none
 */
public fun CharArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Array<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun ByteArray.none(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun ShortArray.none(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun IntArray.none(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun LongArray.none(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun FloatArray.none(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun DoubleArray.none(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> none(map: T -> Int)
 */
public inline fun <T> Array<out T>.onEach(action: (T) -> Unit): Array<out T> {
    return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> onEach(map: T -> Int)
 */
public inline fun ByteArray.onEach(action: (Byte) -> Unit): ByteArray {
    return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> onEach(map: T -> Int)
 */
public inline fun ShortArray.onEach(action: (Short) -> Unit): ShortArray {
    return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> onEach(map: T -> Int)
 */
public inline fun IntArray.onEach(action: (Int) -> Unit): IntArray {
    return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> onEach(map: T -> Int)
 */
public inline fun LongArray.onEach(action: (Long) -> Unit): LongArray {
    return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @sample
 * Array<out T> onEach(map: T -> Int)
 */
public inline fun FloatArray.onEach(action: (Float) ->
Unit): FloatArray {
    return apply { for (element in this) action(element) }\n}\n\n
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.onEach(action: (Double) -> Unit): DoubleArray {
    return apply { for (element in this) action(element) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.onEach(action: (Boolean) -> Unit): BooleanArray {
    return apply { for (element in this) action(element) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.onEachIndexed(action: (Char) -> Unit): CharArray {
    return apply { for (element in this) action(element) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.onEachIndexed(action: (index: Int, T) -> Unit): Array<out T> {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Unit): ByteArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.onEachIndexed(action: (index: Int, Short) -> Unit): ShortArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.onEachIndexed(action: (index: Int, Int) -> Unit): IntArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.onEachIndexed(action: (index: Int, Long) -> Unit): LongArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.onEachIndexed(action: (index: Int, Float) -> Unit): FloatArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.onEachIndexed(action: (index: Int, Double) -> Unit): DoubleArray {
    return apply { forEachIndexed(action) }\n}\n\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }\n}\n
CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply {
        forEachIndexed(action)
    }
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @param sample:samples.collections.Collections.Aggregates.reduce
 */
open inline fun <S, T: S> Array<out T>.reduce(operation: (acc: S, T) -> S): S {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator: S = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
open inline fun ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
open inline fun ShortArray.reduce(operation: (acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
open inline fun IntArray.reduce(operation: (acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
open inline fun LongArray.reduce(operation: (acc: Long, Long) -> Long): Long {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param operation function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
open inline fun FloatArray.reduce(operation: (acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### BooleanArray.reduce

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### CharArray.reduce

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### Array.reduceIndexed

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### ByteArray.reduceIndexed

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### IntArray.reduceIndexed

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.

@sample samples.collections.Collections.Aggregates.reduce

### ByteArray.reduceIndexed

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. `null` if the array is empty. If the array can be empty in an expected way, *please use* [reduceOrIgnoreNull] instead. It returns `null` when its receiver is empty.
Open Source Used In DNAC Platform Arica 7815

```
@sample samples.collections.Collections.Aggregates.reduce

ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    var accumulator = this[0] for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    } return accumulator
}
```

```
IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    var accumulator = this[0] for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    } return accumulator
}
```

```
    var accumulator = this[0] for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    } return accumulator
}
```

```
FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    var accumulator = this[0] for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    } return accumulator
}
```

```
DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
    var accumulator = this[0] for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    } return accumulator
}
```

```
BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {
    var accumulator =
this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

throw UnsupportedOperationException("Empty array can't be reduced.")

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

if (isEmpty())
    throw UnsupportedOperationException("Empty array can't be reduced.")

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator: S = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.

if (isEmpty())
    return null

var accumulator = this[0] for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
}
return accumulator

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element with its index in the original array.
right
* to current accumulator value and each element with its index in the original array.
* Returns `null` if the array is empty.
* * @param [operation] function that takes the index of an element, current accumulator value
and the element itself, * and calculates the next accumulator value.
* * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4")\public inline fun FloatArray.reduceIndexedOrNull(operation: (index: Int, acc: Float, Float) -> Float)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
right\* to current accumulator value and each element with its index in the original array.
* Returns `null` if the array is empty.
* * @param [operation] function that takes the index of an element, current accumulator value
and the element itself, * and calculates the next accumulator value.
* * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4")\public inline fun DoubleArray.reduceIndexedOrNull(operation: (index: Int, acc: Double, Double) -> Double)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
right\* to current accumulator value and each element with its index in the original array.
* Returns `null` if the array is empty.
* * @param [operation] function that takes the index of an element, current accumulator value
and the element itself, * and calculates the next accumulator value.
* * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4")\public inline fun BooleanArray.reduceIndexedOrNull(operation: (index: Int, acc: Boolean, Boolean) -> Boolean)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
right\* to current accumulator value and each element.
* Returns `null` if the array is empty.
* * @param [operation] function that takes current accumulator value and an element, * and calculates
the next accumulator value.
* * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4")\public inline fun CharArray.reduceIndexedOrNull(operation: (index: Int, acc: Char, Char) -> Char)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
right\* to current accumulator value and each element.
* Returns `null` if the array is empty.
* * @param [operation] function that takes current accumulator value and an element, * and calculates
the next accumulator value.
* * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4")\public inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: S, T) -> S)? {\n  if (isEmpty())\n    return null\n  var accumulator: S = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n  }\n  return accumulator
}

*\n@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)\public inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: S, T) -> S)? {\n  if (isEmpty())\n    return null\n  var accumulator: S = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n  }\n  return accumulator
}

*\n@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)\public inline fun ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n  }\n  return accumulator
}

*\n@SinceKotlin("1.4")\@WasExperimental(ExperimentalStdlibApi::class)\public inline fun ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte)? {\n  if (isEmpty())\n    return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n  }\n  return accumulator
}
empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun <S, T : S> Array<out T>.reduceRight(operation: (T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```

empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```

empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```

empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun IntArray.reduceRight(operation: (Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```

empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```

empty. If the array can be empty in an expected way, please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.  

```kotlin
public inline fun FloatArray.reduceRight(operation: (Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

@sample

```kotlin
samples.collections.Collections.Aggregates.reduceRight
```
Accumulates value starting with the last element and applying [operation] from right to left
   * to each element and current accumulator value.
   * Throws an exception if this array is empty. If the array can be empty in an expected way,
   * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
   * @param [operation] function that takes an element and current accumulator value,
   * and calculates the next accumulator value.
   * @sample samples.collections.Collections.Aggregates.reduceRight

public inline fun DoubleArray.reduceRight(operation: (Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun BooleanArray.reduceRight(operation: (Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun CharArray.reduceRight(operation: (Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightIndexed
 */
public inline fun <S, T : S> Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightIndexed
 */
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
expected way.\n* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* \n* \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n*  and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short \n* \nvar index = lastIndex \nif (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\") \nvar accumulator = get(index--) \nwhile (index >= 0) \n{ \n    accumulator = operation(index, get(index), accumulator) \n    --index
} \nreturn accumulator
\n/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int \n* \nvar index = lastIndex \nif (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\") \nvar accumulator = get(index--) \nwhile (index >= 0) \n{ \n    accumulator = operation(index, get(index), accumulator) \n    --index
} \nreturn accumulator
\n/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
LongArray.reduceRightIndexed(operation: (index: Int, Long, acc: Long) -> Long): Long \n* \nvar index = lastIndex \nif (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\") \nvar accumulator = get(index--) \nwhile (index >= 0) \n{ \n    accumulator = operation(index, get(index), accumulator) \n    --index
} \nreturn accumulator
\n/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float \n* \nvar index = lastIndex \nif (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\") \nvar accumulator = get(index--) \nwhile (index >= 0) \n{ \n    accumulator = operation(index, get(index), accumulator) \n    --index
} \nreturn accumulator
\n/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n
DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double \n* \nvar index = lastIndex \nif (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\") \nvar accumulator = get(index--) \nwhile (index >= 0) \n{ \n    accumulator = operation(index, get(index), accumulator) \n    --index
} \nreturn accumulator
\n/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * \n* * Throws an exception if this array is empty. If the array can be empty in an expected way,\n* * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n* * \n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * \n* * @sample

samples.collections.Collections.Aggregates.reduceRight\n

current accumulator value.

* Throws an exception if this array is empty. If the array can be empty in an expected way,
* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean

* var index = lastIndex
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")

* var accumulator = get(index--)
* while (index >= 0) {
*   accumulator = operation(index, get(index), accumulator)
* }

* return accumulator

* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* Returns `null` if the array is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char

* var index = lastIndex
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")

* var accumulator = get(index--)
* while (index >= 0) {
*   accumulator = operation(index, get(index), accumulator)
* }

* return accumulator

* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* Returns `null` if the array is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

ByteArray.reduceRightIndexedOrNull(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte

* var index = lastIndex
* if (index < 0) return null

* var accumulator = get(index--)
* while (index >= 0) {
*   accumulator = operation(index, get(index), accumulator)
* }

* return accumulator

* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* Returns `null` if the array is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

ShortArray.reduceRightIndexedOrNull(operation: (index: Int, short, acc: Short) -> Short): Short

* var index = lastIndex
* if (index < 0) return null

* var accumulator = get(index--)
* while (index >= 0) {
*   accumulator = operation(index, get(index), accumulator)
* }

* return accumulator

* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* Returns `null` if the array is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

IntArray.reduceRightIndexedOrNull(operation: (index: Int, Int, acc: Int) -> Int): Int

* var index = lastIndex
* if (index < 0) return null

* var accumulator = get(index--)
* while (index >= 0) {
*   accumulator = operation(index, get(index), accumulator)
* }

* return accumulator

* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* Returns `null` if the array is empty.

* @param [operation] function that takes the index of an element, the element itself and current accumulator value,
  * and calculates the next accumulator value.

* @sample

samples.collections.Collections.Aggregates.reduceRightOrNull

Open Source Used In DNAC Platform Arica  7822
if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun FloatArray.reduceRightIndexedOrNull(operation: (Int, Float, acc: Float) -> Float): Float? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun DoubleArray.reduceRightIndexedOrNull(operation: (Int, Double, acc: Double) -> Double): Double? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun BooleanArray.reduceRightIndexedOrNull(operation: (Int, Boolean, acc: Boolean) -> Boolean): Boolean? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Array<out T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
return null
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun ByteArray.reduceRightOrNull(operation: (Byte, acc: Byte) -> Byte): Byte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun ShortArray.reduceRightOrNull(operation: (Short, acc: Short) -> Short): Short? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun IntArray.reduceRightOrNull(operation: (Int, acc: Int) -> Int): Int? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun LongArray.reduceRightOrNull(operation: (Long, acc: Long) -> Long): Long? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun FloatArray.reduceRightOrNull(operation: (Float, acc: Float) -> Float): Float? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun DoubleArray.reduceRightOrNull(operation: (Double, acc: Double) -> Double): Double? {
    var index =
lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns `null` if the array is empty or to each element and current accumulator value.
 * and calculates the next accumulator value.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun BooleanArray.reduceRightOrNull(operation: (Boolean, acc: Boolean) -> Boolean): Boolean? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from right to left
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 */
@SinceKotlin("1.4")
public inline fun <T, R> Array<out T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.runningFold(initial: R, operation: (acc: R, Byte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.runningFold(initial: R, operation: (acc: R, Short) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
[operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold

Result adicionado[operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```
operation: (acc: R, Char) -> R): List<R> { ln if (isEmpty()) return listOff(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (element in this) { \n        accumulator = operation(accumulator, element)\n        result.add(accumulator)\n    }\n    return result\n}/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\npublic inline fun <T, R> Array<out T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> { ln if (isEmpty()) return listOff(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) { \n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> { ln if (isEmpty()) return listOff(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) { \n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun <R> ShortArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> { ln if (isEmpty()) return listOff(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) { \n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * and the element itself, and calculates the next accumulator value.\n */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun <R> IntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): List<R> { ln if (isEmpty()) return listOff(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) { \n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}/**
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
    }
    result.add(accumulator)
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
FloatArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
    }
    result.add(accumulator)
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
DoubleArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
    }
    result.add(accumulator)
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
BooleanArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
    }
    result.add(accumulator)
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
CharArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    if (isEmpty())
        return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
    }
    result.add(accumulator)
    return result
}
takes current accumulator value and the element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.runningReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.runningReduce(operation: (acc: Short, Short) -> Short): List<Short> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.runningReduce(operation: (acc: Int, Int) -> Int): List<Int> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun LongArray.runningReduce(operation: (acc: Long, Long) -> Long): List<Long> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Long>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.runningReduce(operation: (acc: Float, Float) -> Float): List<Float> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Float>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    Array<out T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <S, T : S>
    List<S>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<S>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

The `runningReduce` function takes current accumulator value and an element, and calculates the next accumulator value.

* Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.4")
@kotlin.experimental.ExperimentalStdlibApi
public inline fun <S, T : S>
    List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.runningReduce(operation: (acc: Double, Double) -> Double): List<Double> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Double>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.runningReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Boolean>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Array<out T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.runningReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of this array.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.runningReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
add(accumulator) } for (index in 1 until size) {\n    accumulator = operation(index, accumulator, this[index])\n    result.add(accumulator)\n}\n\nreturn result
\n\n*\n@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\n
public inline fun IntArray.runningReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Int>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result
\n}
each element, its index in the original array and current accumulator value that starts with the first element of this
array.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@public inline fun CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to
each element and current accumulator value that starts with [initial] value.

 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value
in resulting list.

 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@public inline fun CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to
each element and current accumulator value that starts with [initial] value.

 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value
in resulting list.

 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@public inline fun CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
not be mutated; otherwise it would affect the previous value in resulting list.

- **@param [operation]** function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value. The index in the original array and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. The index in the element, current accumulator value and the element itself, and calculates the next accumulator value.

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
public inline fun <R> ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
    return runningFoldIndexed(initial, operation)
}
```

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
    return runningFoldIndexed(initial, operation)
}
```

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
public inline fun <R> FloatArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
public inline fun <R> DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

- **@sample samples.collections.Collections.Aggregates.scan**

```kotlin
public inline fun <R> BooleanArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

---

**Open Source Used In DNAC Platform Arica 7834**
not be mutated; \n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.scan\n
```kotlin
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n}
```

```kotlin
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\npublic inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n```
c inline fun ByteArray.scanReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed instead.",
    ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun ShortArray.scanReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed instead.",
    ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun IntArray.scanReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed
    instead.", ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed
    instead.", ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun FloatArray.scanReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed
    instead.", ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun DoubleArray.scanReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed
    instead.", ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun BooleanArray.scanReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    return runningReduceIndexed(operation)
}

@Deprecated("Use runningReduceIndexed
    instead.", ReplaceWith("runningReduceIndexed(operation)"), level =
    DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public <T> Array<out T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function
 * applied to each element in the array.
 */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public <T> Array<out T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumByDouble(selector: (Short) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumByDouble(selector: (Float) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun ByteArray.sumOf(selector: (Byte) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun ShortArray.sumOf(selector: (Short) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun IntArray.sumOf(selector: (Int) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
Open Source Used In DNAC Platform Arica 7838
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.sumOf(selector: (Long) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\npublic inline fun
FloatArray.sumOf(selector: (Float) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\npublic inline fun
DoubleArray.sumOf(selector: (Double) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\npublic inline fun
CharArray.sumOf(selector: (Char) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\npublic inline fun
ShortArray.sumOf(selector: (Short) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\npublic inline fun
IntArray.sumOf(selector: (Int) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.sumOf(selector: (Long) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
function applied to each element in the array.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun FloatArray.sumOf(selector: (Float) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun DoubleArray.sumOf(selector: (Double) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun BooleanArray.sumOf(selector: (Boolean) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun CharArray.sumOf(selector: (Char) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun <T> Array<out T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun ByteArray.sumOf(selector: (Byte) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun ShortArray.sumOf(selector: (Short) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun IntArray.sumOf(selector: (Int) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun LongArray.sumOf(selector: (Long) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

Open Source Used In DNAC Platform Arica  7840
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun FloatArray.sumOf(selector: (Float) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun DoubleArray.sumOf(selector: (Double) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun BooleanArray.sumOf(selector: (Boolean) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun CharArray.sumOf(selector: (Char) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun <T> Array<out T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun ByteArray.sumOf(selector: (Byte) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun ShortArray.sumOf(selector: (Short) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun IntArray.sumOf(selector: (Int) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun LongArray.sumOf(selector: (Long) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
public inline fun FloatArray.sumOf(selector: (Float) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("sumOfULong")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun DoubleArray.sumOf(selector: (Double) -> ULong): ULong { \n    var sum: ULong = 0.toULong(); \n    for (element in this) { \n        sum += selector(element); \n    } \n    return sum
} \n
/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("sumOfULong")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun BooleanArray.sumOf(selector: (Boolean) -> ULong): ULong { \n    var sum: ULong = 0.toULong(); \n    for (element in this) { \n        sum += selector(element); \n    } \n    return sum
} \n
/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.
 */
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.jvm.JvmName("sumOfULong")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun CharArray.sumOf(selector: (Char) -> ULong): ULong { \n    var sum: ULong = 0.toULong(); \n    for (element in this) { \n        sum += selector(element); \n    } \n    return sum
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public fun <T : Any> Array<T?>.requireNoNulls(): Array<T> { \n    for (element in this) { \n        if (element == null) { \n            throw IllegalArgumentException("null element found in $this.")
        }
    } \n    @Suppress("UNCHECKED_CAST")\n    return this as Array<T>
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public inline fun <T> Array<out T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> { \n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) { \n        if (predicate(element)) { \n            first.add(element)
        } else { \n            second.add(element)
        }
    } \n    return Pair(first, second)
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public inline fun ByteArray.partition(predicate: (Byte) -> Boolean): Pair<List<Byte>, List<Byte>> { \n    val first = ArrayList<Byte>()\n    val second = ArrayList<Byte>()\n    for (element in this) { \n        if (predicate(element)) { \n            first.add(element)
        } else { \n            second.add(element)
        }
    } \n    return Pair(first, second)
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public inline fun ShortArray.partition(predicate: (Short) -> Boolean): Pair<List<Short>, List<Short>> { \n    val first = ArrayList<Short>()\n    val second = ArrayList<Short>()\n    for (element in this) { \n        if (predicate(element)) { \n            first.add(element)
        } else { \n            second.add(element)
        }
    } \n    return Pair(first, second)
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public inline fun IntArray.partition(predicate: (Int) -> Boolean): Pair<List<Int>, List<Int>> { \n    val first = ArrayList<Int>()\n    val second = ArrayList<Int>()\n    for (element in this) { \n        if (predicate(element)) { \n            first.add(element)
        } else { \n            second.add(element)
        }
    } \n    return Pair(first, second)
} \n
/**
 * Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`. \n */
public inline fun LongArray.partition(predicate: (Long) -> Boolean): Pair<List<Long>, List<Long>> { \n    val first = ArrayList<Long>()\n    val second = ArrayList<Long>()\n    for (element in this) { \n        if (predicate(element)) { \n            first.add(element)
        } else { \n            second.add(element)
        }
    } \n    return Pair(first, second)
} \n
Open Source Used In DNAC Platform Arica 7842
Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, while *second* list contains elements for which [predicate] yielded `false`. *first* list contains elements for which [predicate] yielded `true`, while *second* list contains elements for which [predicate] yielded `false`. Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, while *second* list contains elements for which [predicate] yielded `false`. Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, while *second* list contains elements for which [predicate] yielded `false`. Splits the original array into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, while *second* list contains elements for which [predicate] yielded `false`. Returns a list of pairs built from the elements of `this` array and the [other] array with the same index. The returned list has length of the shortest collection. Returns a list of pairs built from the elements of `this` array and the [other] array with the same index. The returned list has length of the shortest collection. Returns a list of pairs built from the elements of `this` array and the [other] array with the same index. The returned list has length of the shortest collection. Returns a list of pairs built from the elements of `this` array and the [other] array with the same index. The returned list has length of the shortest collection.
R>): List<Pair<Double, R>> { t1, t2 -> t1 to t2 }\n\n* Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable\n\n/**\n * Returns a list of values built from the elements of `this` array and the [other] array with the same index\n* The returned list has length of the shortest collection.\n* @sample samples.collections.Operetions.zipIterable
with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val valSize = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {
    val valSize = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

* Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.

@sample

samples.collections.Operations.zipIterableWithTransform

public inline fun <R, V> Array.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val valArraySize = size
    val list = ArrayList<V>(size)
    var i = 0
    for (element in other) {
        list.add(transform(element, other[i]))
        i = i + 1
    }
    return list
}
if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

*<R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> { val arraySize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
 var i = 0
 for (element in other) {
    if (i >= arraySize) break
    list.add(transform(this[i++], element))
  }
  return list
}* Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.

*/

public inline fun <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

*<R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> { val arraySize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
 var i = 0
 for (element in other) {
    if (i >= arraySize) break
    list.add(transform(this[i++], element))
  }
  return list
}* Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.

*/

    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

*<R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> { val arraySize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
 var i = 0
 for (element in other) {
    if (i >= arraySize) break
    list.add(transform(this[i++], element))
  }
  return list
}* Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.

*/

public inline fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

*<R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> { val arraySize = size
 val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
 var i = 0
 for (element in other) {
    if (i >= arraySize) break
    list.add(transform(this[i++], element))
  }
  return list
}* Returns a list of values built from the elements of 'this' array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.

*/

Open Source Used In DNAC Platform Arica  7846
shortest collection.
* The returned list has length of the shortest collection.

```kotlin
*public inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> { val arraySize = size
val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
for (element in other) {
    list.add(transform(element, this))
}
return list

*public inline fun <R, V> CharArray.zipIterableWithTransformfun CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> { val arraySize = size
val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
for (element in other) {
    list.add(transform(element, this))
}
return list

*public inline fun <R, V> CharArray.zipIterableWithTransformfun CharArray.zipIterable(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> { val arraySize = size
val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
for (element in other) {
    list.add(transform(element, this))
}
return list
```

```kotlin
*public inline fun <R, V> CharArray.zipIterableWithTransformfun CharArray.zipIterable(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> { val arraySize = size
val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
for (element in other) {
    list.add(transform(element, this))
}
return list
```
ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same
 * index using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zip
 */
public inline fun <V> LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zip
 */
public inline fun <V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zip
 */
public inline fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zip
 */
public inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zip
 */
public inline fun <V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Eppends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <T, A : Appendable> Array<out T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
if (limit >= 0 && count > limit) buffer.append(truncated)
buffer.append(postfix)
return
buffer.append(transform(element)))
else
buffer.append(element.toString())
} else break
}
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. 
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
 * defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> ShortArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? = null): A {
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator)
if (limit < 0 || count <= limit) {
if (transform != null)
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. 
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
 * defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> IntArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): A {
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator)
if (limit < 0 || count <= limit) {
if (transform != null)
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. 
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
 * defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): A {
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator)
if (limit < 0 || count <= limit) {
if (transform != null)
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. 
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
 * defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): A {
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator)
if (limit < 0 || count <= limit) {
if (transform != null)
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}
buffer.append(transform(element))
else
buffer.append(element.toString())
} else break
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. 
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
 * defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): A {
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator) \n  if (limit < 0 || count <= limit) \n      return buffer
    \n    buffer.append(transform(element)) \n    else
    \n    buffer.append(element.toString()) \n  } else break

\n  if (limit > 0 && count > limit) buffer.append(truncated) \n  buffer.append(postfix) \n  return buffer
}

buffer.append(separator) \n  if (limit < 0 || count <= limit) \n      return buffer
    \n    buffer.append(transform(element)) \n    else
    \n    buffer.append(element.toString()) \n  } else break

\n  if (limit > 0 && count > limit) buffer.append(truncated) \n  buffer.append(postfix) \n  return buffer
}

buffer.append(separator) \n  if (limit < 0 || count <= limit) \n      return buffer
    \n    buffer.append(transform(element)) \n    else
    \n    buffer.append(element.toString()) \n  } else break

\n  if (limit > 0 && count > limit) buffer.append(truncated) \n  buffer.append(postfix) \n  return buffer
}

buffer.append(separator) \n  if (limit < 0 || count <= limit) \n      return buffer
    \n    buffer.append(transform(element)) \n    else
    \n    buffer.append(element.toString()) \n  } else break

\n  if (limit > 0 && count > limit) buffer.append(truncated) \n  buffer.append(postfix) \n  return buffer
}

buffer.append(separator) \n  if (limit < 0 || count <= limit) \n      return buffer
    \n    buffer.append(transform(element)) \n    else
    \n    buffer.append(element.toString()) \n  } else break

\n  if (limit > 0 && count > limit) buffer.append(truncated) \n  buffer.append(postfix) \n  return buffer
}
CharSequence = ",", prefix: CharSequence = ",", postfix: CharSequence = ",", limit: Int = -1, truncated: 
CharSequence = ",", transform: ((Int) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString().
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix]
 * if supplied.\n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only
 * the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "\n...\n").\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Array<out T>.asIterable(): Iterable<T> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> ByteArray.asIterable(): Iterable<Byte> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> ShortArray.asIterable(): Iterable<Short> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> IntArray.asIterable(): Iterable<Int> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> LongArray.asIterable(): Iterable<Long> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> BooleanArray.asIterable(): Iterable<Boolean> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.\n * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> CharArray.asIterable(): Iterable<Char> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}
(isEmpty()) return emptyList();
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun FloatArray.asIterable(): Iterable<Float> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun DoubleArray.asIterable(): Iterable<Double> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun BooleanArray.asIterable(): Iterable<Boolean> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun CharArray.asIterable(): Iterable<Char> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun ByteArray.asSequence(): Sequence<Byte> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun ShortArray.asSequence(): Sequence<Short> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun IntArray.asSequence(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun FloatArray.asSequence(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun DoubleArray.asSequence(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun BooleanArray.asSequence(): Sequence<Boolean> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun CharArray.asSequence(): Sequence<Char> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Array<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Array<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Array<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Array<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
Double = 0.0
var count: Int = 0
for (element in this) {
    sum += element
    ++count
}
return if (count == 0) Double.NaN else sum / count

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("average")
public fun Array<out Any>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Array<out Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Array<out Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ByteArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ShortArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun IntArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun LongArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun FloatArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun DoubleArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sum")
public fun Array<out Any>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Array<out Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Array<out Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Array<out Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Array<out Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Array<out Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Array<out Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

*/

Open Source Used In DNAC Platform Arica 7853
license/LICENSE.txt file

/*@

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongRange.random(): Long {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun CharRange.random(): Char {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public fun IntRange.random(random: Random): Int {
    try {
        return random.nextInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public fun LongRange.random(random: Random): Long {
    try {
        return random.nextLong(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public fun CharRange.random(random: Random): Char {
    try {
        return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun IntRange.randomOrNull(): Int? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun LongRange.randomOrNull(): Long? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun CharRange.randomOrNull(): Char? {
    if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Returns `true` if this range contains the specified 
 * element. Always returns `false` if the 
 * element is `null`.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline operator fun IntRange.contains(element: Int?): Boolean {
    return if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}
Checks if the specified [value] belongs to this range.\n/*\n@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n@DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n@kotlin.jvm.JvmName("longRangeContains")\npublic operator fun ClosedRange<Long>.contains(value: Float): Boolean {
    return value.toLongExactOrNull().let { if (it != null) contains(it) else false }\n    \n    *\n
    \n    \n    *\n
    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("byteRangeContains")\n    public operator fun ClosedRange<Byte>.contains(value: Float): Boolean {
        return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("shortRangeContains")\n    public operator fun ClosedRange<Short>.contains(value: Float): Boolean {
        return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n    }

    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("doubleRangeContains")\n    public operator fun ClosedRange<Double>.contains(value: Float): Boolean {
        return contains(value.toDouble())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("longRangeContains")\n    public operator fun ClosedRange<Long>.contains(value: Int): Boolean {
        return contains(value.toLong())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("byteRangeContains")\n    public operator fun ClosedRange<Byte>.contains(value: Int): Boolean {
        return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("shortRangeContains")\n    public operator fun ClosedRange<Short>.contains(value: Int): Boolean {
        return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("doubleRangeContains")\n    public operator fun ClosedRange<Double>.contains(value: Int): Boolean {
        return contains(value.toDouble())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("floatRangeContains")\n    public operator fun ClosedRange<Float>.contains(value: Int): Boolean {
        return contains(value.toFloat())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("intRangeContains")\n    public operator fun ClosedRange<Int>.contains(value: Long): Boolean {
        return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("byteRangeContains")\n    public operator fun ClosedRange<Byte>.contains(value: Long): Boolean {
        return value.toByteExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("shortRangeContains")\n    public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
        return value.toShortExactOrNull().let { if (it != null) contains(it) else false }\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("doubleRangeContains")\n    public operator fun ClosedRange<Double>.contains(value: Long): Boolean {
        return contains(value.toDouble())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("floatRangeContains")\n    public operator fun ClosedRange<Float>.contains(value: Long): Boolean {
        return contains(value.toFloat())\n    }

    \n    */

    @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.\")\n    @DeprecatedSinceKotlin(warningSince = "1.3.0", errorSince = "1.4.0")\n    @kotlin.jvm.JvmName("intRangeContains")\n    public operator fun ClosedRange<Int>.contains(value: Long): Boolean {
        return value.toIntExactOrNull().let { if (it != null) contains(it) else false }\n    }

    */

    Open Source Used In DNAC Platform Arica 7856
and floating point arguments has ambiguous semantics and is going to be removed."
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Long): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Short): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Returns a progression from this value down to
 * the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@kotlin.jvm.InfixOperator(true)
@kotlin.jvm.JvmName("IntProgression.fromClosedRange")
public infix fun IntProgression.fromClosedRange(from: Int, to: Int, step: Int): IntProgression {
    return IntProgression(from, to, step)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@kotlin.jvm.InfixOperator(true)
@kotlin.jvm.JvmName("LongProgression.fromClosedRange")
public infix fun LongProgression.fromClosedRange(from: Long, to: Long, step: Long): LongProgression {
    return LongProgression(from, to, step)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@kotlin.jvm.InfixOperator(true)
@kotlin.jvm.JvmName("ByteProgression.fromClosedRange")
public infix fun ByteProgression.fromClosedRange(from: Byte, to: Byte, step: Byte): ByteProgression {
    return ByteProgression(from, to, step)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@kotlin.jvm.InfixOperator(true)
@kotlin.jvm.JvmName("CharProgression.fromClosedRange")
public infix fun CharProgression.fromClosedRange(from: Char, to: Char, step: Char): CharProgression {
    return CharProgression(from, to, step)
}
value is greater than 'this' value the returned progression is empty.

*\public infix fun Short.downTo(to: Int): IntProgression { return IntProgression.fromClosedRange(this.toInt(), to, -1L)\n//**\n
\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.\n\* Returns a progression from this value down to the specified [to] value with the step -1.\n\* The [to] value should be less than or equal to 'this' value.\n\* If the [to] value is greater than 'this' value the returned progression is empty.
Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null
}

internal fun Double.toByteExactOrNull(): Byte? {
    return if (this in
    Double.MIN_VALUE.toDouble()..Double.MAX_VALUE.toDouble()) this.toInt().toByte() else null
}

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in
    Float.MIN_VALUE.toFloat()..Float.MAX_VALUE.toFloat()) this.toInt().toByte() else null
}

internal fun Long.toIntExactOrNull(): Int? {
    return if (this in
    Long.MIN_VALUE.toLong()..Long.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in
    Double.MIN_VALUE.toDouble()..Double.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun "Open Source Used In DNAC Platform Arica 7859"
[to] value is less than or equal to `this` value, then the returned range is empty.

```kotlin
public infix fun Long.until(to: Long): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this .. (to - 1).toLong()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Byte.until(to: Long): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this .. (to - 1).toLong()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Short.until(to: Long): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this .. (to - 1).toLong()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Int.until(to: Short): IntRange {
    return this .. (to.toInt() - 1).toInt()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Long.until(to: Short): LongRange {
    return this .. (to.toLong() - 1).toLong()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Byte.until(to: Short): IntRange {
    return this.toInt() .. (to.toInt() - 1).toInt()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public infix fun Short.until(to: Short): IntRange {
    return this.toInt() .. (to.toInt() - 1).toInt()
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun <T : Comparable<T>> T.coerceAtLeast(minimumValue: T): T {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Byte.coerceAtLeast(minimumValue: Byte): Byte {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Short.coerceAtLeast(minimumValue: Short): Short {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Int.coerceAtLeast(minimumValue: Int): Int {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Long.coerceAtLeast(minimumValue: Long): Long {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Float.coerceAtLeast(minimumValue: Float): Float {
    return if (this < minimumValue) minimumValue else this
}
```

Returns a range from this value up to but excluding the specified [to] value.

```kotlin
public fun Double.coerceAtLeast(minimumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun <T : Comparable<T>> T.coerceAtMost(maximumValue: T): T {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Byte.coerceAtMost(maximumValue: Byte): Byte {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Short.coerceAtMost(maximumValue: Short): Short {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Float.coerceAtMost(maximumValue: Float): Float {
    return if (this > maximumValue) maximumValue else this
}
```

Ensures that this value is not less than the specified [minimumValue].

```kotlin
public fun Double.coerceAtMost(maximumValue: Double): Double {
    return if (this > maximumValue) maximumValue else this
}
```
* Ensures that this value is not greater than the specified [maximumValue].

```kotlin
public fun Byte.coerceAtMost(maximumValue: Byte): Byte {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Short.coerceAtMost(maximumValue: Short):
Long.coerceAtMost(maximumValue: Long) {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInComparable
 */
public fun <T : Comparable<T>> T.coerceIn(minimumValue: T?, maximumValue: T?): T {
    if (minimumValue !== null && maximumValue !== null) {
        if (minimumValue > maximumValue) throw
            IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
        if (this < minimumValue) return minimumValue
        if (this > maximumValue) return maximumValue
    }
    else {
        if (minimumValue !== null && this < minimumValue) return minimumValue
        if (maximumValue !== null && this > maximumValue) return maximumValue
    }
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Byte.coerceIn(minimumValue: Byte,
maximumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw
            IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
        if (this < minimumValue) return minimumValue
        if (this > maximumValue) return maximumValue
    }
    return this
}
```
is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

```kotlin
@sample
samples.comparisons.ComparableOps.coerceIn

public fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue], [maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue], [maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue], [maximumValue].
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 
 * 
 * @sample samples.comparisons.ComparableOps.coerceInFloatingPointRange
 */
@SinceKotlin("1.1")
public fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        range.lessThanOrEquals(this, range.start) && !range.lessThanOrEquals(range.start, this) -> range.start
        !range.lessThanOrEquals(range.start, this) -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue], [maximumValue].
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 
 * 
 * @sample samples.comparisons.ComparableOps.coerceInComparable
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}
```

```kotlin
@sample
samples.comparisons.ComparableOps.coerceIn

public fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified 
 * range [minimumValue], [maximumValue].
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`. 
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Int.coerceIn(range: ClosedRange<Int>): Int {
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}
```
specified [range].

* @return this value if it's in the [range], or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.

* @sample

```java
public fun Long.coerceIn(range: ClosedRange<Long>): Long {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Long>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return
    when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else ->
            this
    }
}
```
the same as the bits of this `UByte` value,
* whereas the most significant 24 bits are filled with zeros.

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFF) /* * Converts this [UByte] value to [ULong].
* The resulting `ULong` value represents the same numerical value as this `UByte`.
* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte` value,
* whereas the most significant 56 bits are filled with zeros.

@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = ULong(data.toLong() and 0xFF) /* * Converts this [UByte] value to [Float].
* The resulting `Float` value represents the same numerical value as this `UByte`.
* @kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toInt().toFloat() /* * Converts this [UByte] value to [Double].
* The resulting `Double` value represents the same numerical value as this `UByte`.
* If this value is positive, the resulting `UByte` value represents the same numerical value as this `Byte`.
* The resulting `UByte` value has the same binary representation as this `Byte` value.

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UInt @PublishedApi internal constructor(@PublishedApi internal val data: Int) : Comparable<UInt> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UInt can have.
         */
        public const val MIN_VALUE: UInt = UInt(0)

        /**
         * A constant holding the maximum value an instance of UInt can have.
         */
        public const val MAX_VALUE: UInt = UInt(-1)

        /**
         * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
         */

        /**
         * The number of bytes used to represent an instance of UInt in a binary form.
         */
        public const val SIZE_BYTES: Int = 4

        /**
         * The number of bits used to represent an instance of UInt in a binary form.
         */
        public const val SIZE_BITS: Int = 32
    }

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
     * or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline fun compareTo(other: UByte): Int = this.compareTo(other.toUInt()) /* * Compares this value with the specified value for order.
* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
* or a positive number if it's greater than other.
* @kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt()) /* * Compares this value with the specified value for order.
* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
* or a positive number if it's greater than other.
* @kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt()) /* * Compares this value with the specified value for order.
value, a negative number if it's less than other, a positive number if it's greater than other.

```kotlin
@kotlin.internal.InlineOnly
@Suppress("OVERRIDE_BY_INLINE")
public override inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number
 * if it's less than other, a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())

/**
 * Adds the other value to this value. * 
 * @kotlin.internal.InlineOnly
 * @kotlin.internal.InlineOnly
 * @kotlin.internal.InlineOnly
 * @kotlin.internal.InlineOnly
 */
```
other.data) /* Performs a bitwise XOR operation between the two values. */ */ public inline infix fun xor(other: UInt): UInt = UInt(this.data xor other.data) /* Inverts the bits in this value. */ */ public inline fun.inv(): UInt = UInt(data.inv()) /* Converts this [UInt] value to [Byte]. */ */ If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents the same numerical value as this `UInt` value. */ */ The resulting `Byte` value is represented by the least significant 8 bits of this `UInt` value. */ */ Note that the resulting `Byte` value may be negative. */ */ public inline fun.toByte(): Byte = data.toByte() /* Converts this [UInt] value to [Short]. */ */ If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents the same numerical value as this `UInt` value. */ */ The least significant 16 bits of this `UInt` value are the same as the bits of this `Short` value. */ */ Note that the resulting `Short` value may be negative. */ */ public inline fun.toShort(): Short = data.toShort() /* Converts this [UInt] value to [Int]. */ */ If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int` value represents the same numerical value as this `UInt`. Otherwise the result is negative. */ */ The resulting `Int` value has the same binary representation as this `UInt` value. */ */ public inline fun.toInt(): Int = data /* Converts this [UInt] value to [Long]. */ */ The least significant 32 bits of the resulting `Long` value are the same as the bits of this `UInt` value, whereas the most significant 32 bits are filled with zeros. */ */ The resulting `Long` value represents the same numerical value as this `UInt` value. */ */ The least significant 32 bits of this `UInt` value are the same as the bits of this `Long` value, whereas the most significant 32 bits are filled with zeros. */ */ Returns this value. */ */ public inline fun.toUByte(): UByte = data.toUByte() /* Converts this [UInt] value to [UByte]. */ */ If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `UInt`. */ */ The resulting `UByte` value represents the least significant 8 bits of this `UInt` value. */ */ The least significant 32 bits of this ` UInt` value are the same as the bits of this `UByte` value, whereas the most significant 32 bits are filled with zeros. */ */ Converts this [UInt] value to [UByte]. */ */ If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `UInt`. */ */ The resulting `UByte` value is represented by the least significant 8 bits of this `UInt` value. */ */ The least significant 32 bits of this `UInt` value are the same as the bits of this `UShort` value, whereas the most significant 32 bits are filled with zeros. */ */ The resulting `UShort` value represents the same numerical value as this `UInt` value. */ */ The resulting `UShort` value is represented by the least significant 16 bits of this `UInt` value. */ */ The least significant 16 bits of this `UInt` value are the same as the bits of this `UShort` value, whereas the most significant 16 bits are filled with zeros. */ */ Converts this [UInt] value to [UShort]. */ */ Note that the resulting `UShort` value may be negative. */ */ public inline fun.toUShort(): UShort = data.toUShort() /* Converts this [UInt] value to [ULong]. */ */ The resulting `ULong` value represents the same numerical value as this `UInt`. */ */ The least significant 32 bits of this `UInt` value are the same as the bits of this `ULong` value, whereas the most significant 32 bits are filled with zeros. */ */ Converts this [UInt] value to [ULong]. */ */ The resulting value is the closest `Float` to this `UInt` value. */ */ In case when this `UInt` value is exactly between two `Float` s, the one with zero at least significant bit of mantissa is selected. */ */ Converts this [UInt] value to [Double]. */ */ The resulting value is the closest `Double` to this `UInt` value. */ */ In case when this `UInt` value is exactly between two `Double` s, the one with the smallest exponent is selected. */ */ Converts this [UInt] value to [Float]. */ */ The resulting value is the closest `Float` to this `UInt` value. */ */ The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `Byte` value, whereas the most significant 24 bits are filled with the sign bit of this value. */ */ Since Kotlin("1.3") */ */ Converts this [UInt] value to [Byte]. */ */ Converts this [Short] value to [UInt]. */ */ If this value is positive, the resulting `UInt` value represents the same numerical value as this `Short` value. */ */ The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value, whereas the most significant 16 bits are filled with the sign bit of this value. */ */ Since Kotlin("1.3") */ */ Converts this [Int] value to [UInt]. */ */ If this value is positive, the resulting `UInt` value represents the same numerical value as this `Int` value. */ */ The resulting `UInt` value has the
same binary representation as this `Int` value.

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
\n\npublic inline fun Int.toUInt(): UInt = UInt(this)
/**
 * Converts this [Long] value to [UInt].\n * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Long`.
 * The resulting `UInt` value is represented by the least significant 32 bits of this `Long` value.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
\n\npublic inline fun Long.toUInt(): UInt = UInt(this.toInt())

/**
 * Converts this [Float] value to [UInt].
 *
 * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Float`.
 *
 * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Float` value is negative or `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE] \n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
\n\npublic inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())
/**
 * Converts this [Double] value to [UInt].
 *
 * The fractional part, if any, is rounded down towards zero.\n * Returns zero if this `Double` value is negative or `NaN`, [UInt.MAX_VALUE] if it's bigger than [UInt.MAX_VALUE].
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
\n\npublic inline fun Double.toUInt(): UInt = doubleToUInt(this)


---

Open Source Used In DNAC Platform Arica 7868
UInt = this.toUInt().minus(other.toUInt()) /** Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)
/** Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
/** Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())
/** Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())
/** Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)
/** Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
/** Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())
/** Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())
/** Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)
/** Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong = this.toULong().div(other)
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
/** Creates a range from this value to the specified [other] value. */
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange = UIntRange(this.toUInt(), other.toUInt())
/** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: UShort): UShort = UShort(this.data and other.data)
/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: UShort): UShort = UShort(this.data or other.data)
/** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: UShort): UShort = UShort(this.data xor other.data)
/** Inverts the bits in this value. */
@kotlin.internal.InlineOnly
public inline fun inv(): UShort = UShort(data.inv())
/** Converts this [UShort] value to [Byte]. */
@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()
/** Converts this [UShort] value to [Short]. */
@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data
/** Converts this [UShort] value to [Int]. */
@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt() and 0xFFFF
/** Converts this [UShort] value to [Long]. */
@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong() and 0xFFFFFFFF
/** Converts this [UShort] value to [Double]. */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = data.toDouble()
of this `UShort` value, * whereas the most significant 48 bits are filled with zeros.

@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = data.toLong() and 0xFFFF /* ** Converts this [ULong] value to [UShort].

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte() /* ** Returns this value.

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFFFF) /* ** Converts this [UShort] value to [UInt].

@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF) /* ** Converts this [UShort] value to [ULong].

@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toInt().toFloat() /* Converts this `UShort` value to `Float`.

@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = this.toInt().toDouble() /* Converts this `UShort` value to `Double`.

override fun toString(): String = toInt().toString() /* Converts this `UShort` value to `String`.

fun Byte.toUShort(): UShort = this.toShort() /* Converts this `Byte` value to `UShort`.

fun Short.toUShort(): UShort = this /* Converts this `Short` value to `UShort`.

fun Int.toUShort(): UShort = this.toShort() /* Converts this `Int` value to `UShort`.

fun Long.toUShort(): UShort = this.toShort() /* Converts this `Long` value to `UShort`.

@SinceKotlin("1.3") @ExperimentalUnsignedTypes
public inline fun CharRange.startInclusive: Char get() = first /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

@SinceKotlin("1.3") @ExperimentalUnsignedTypes
public inline fun CharRange.endInclusive: Char get() = last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun contains(value: Char): Boolean = first <= value && value <= last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun isEmpty(): Boolean = first > last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun equals(other: Any?): Boolean = other is CharRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun hashCode(): Int = if (isEmpty()) -1 else (first.toInt() * 31 + last.toInt()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun toString(): String = "$first..$last" /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {

override val first: Char get() = start /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override val last: Char get() = endInclusive /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun contains(value: Char): Boolean = first <= value && value <= last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun isEmpty(): Boolean = first > last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun equals(other: Any?): Boolean = other is CharRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun hashCode(): Int = if (isEmpty()) -1 else (first.toInt() * 31 + last.toInt()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

override fun toString(): String = "$first..$last" /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.startInclusive: Char get() = first /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.endInclusive: Char get() = last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.contains(value: Char): Boolean = first <= value && value <= last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.isEmpty(): Boolean = first > last /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.equals(other: Any?): Boolean = other is CharRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.hashCode(): Int = if (isEmpty()) -1 else (first.toInt() * 31 + last.toInt()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

fun CharRange.toString(): String = "$first..$last" /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.

CharRange(1.toChar(), 0.toChar()) /* The resulting `CharRange` value is the same as the bits of this `CharRange` value.
range of values of type 'Int'.

An empty range of values of type Int.

An empty range of values of type Long.

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

* * *
List<E>

/**
 * Returns an [IntRange] of the valid indices for this collection.
 * @sample samples.collections.Collections.Collections.indicesOfCollection
 */
public val Collection<*>.indices: IntRange
    get() = 0..size - 1

/**
 * Returns the index of the last item in the list or -1 if the list is empty.
 * @sample samples.collections.Collections.Collections.lastIndexOfCollection
 */
public val <T> List<T>.lastIndex: Int
    get() = size - 1

/**
 * Returns `true` if the collection is not empty.
 * @sample samples.collections.Collections.Collections.isCollectionNotEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> Collection<T>.isNotEmpty(): Boolean = !isEmpty()

/**
 * Returns the index of the last item in the list or -1 if the list is empty.
 * @sample samples.collections.Collections.Collections.lastIndexOfList
 */
public val <T> List<T>.lastIndex: Int
    get() = size - 1

/**
 * Returns `true` if this nullable collection is
 * either null or empty.
 * @sample samples.collections.Collections.Collections.isNullOrEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isNullOrEmpty(): Boolean = this == null || this.isEmpty()

/**
 * Returns this Collection if it's not `null` and the empty list otherwise.
 * @sample samples.collections.Collections.Collections.emptyListIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Collection<*>, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Checks if all elements in the specified collection are contained in this collection.
 * @sample samples.collections.Collections.Collections.collectionContainsAll
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)

/**
 * Searches this list or its range for the provided [element] using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the
 * Comparable natural ordering of its elements, otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * otherwise the result is undefined.
 * The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.
 */
@sample samples.collections.Collections.Collections.Lists.binarySearchOnComparable

insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted according to the specified comparator.

```kotlin
@sample samples.collections.Collections.Lists.binarySearchWithComparator

List<T>.binarySearch(element: T, comparator: Comparator<? super T>, fromIndex: Int = 0, toIndex: Int = size): Int {
    var low = fromIndex
    var high = toIndex - 1
    while (low <= high) {
        var mid = (low + high).ushr(1)
        var midVal = get(mid)
        val cmp = comparator.compare(midVal, element)
        if (cmp < 0) low = mid + 1
        else if (cmp > 0) high = mid - 1
        else return mid // key found
    }
    return -(low + 1)  // key not found
}
```

The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements. If the list contains multiple elements with the specified key, there is no guarantee which one will be found.

```kotlin
@sample samples.collections.Collections.Lists.binarySearchByKey

List<T>.binarySearchBy(key: K?, fromIndex: Int = 0, toIndex: Int = size, crossinline selector: (T) -> K?): Int =
    binarySearch(fromIndex, toIndex) { compareValues(selector(it), key) }
```

The result is undefined. If the list contains multiple elements with the specified key, it is contained in the list within the specified range, otherwise, the inverted insertion point `(insertion point - 1)`.

```kotlin
@sample samples.collections.Collections.Lists.binarySearchWithComparisonFunction

List<T>.binarySearch(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int {
    var low = fromIndex
    var high = toIndex - 1
    while (low <= high) {
        var mid = (low + high).ushr(1)
        var midVal = get(mid)
        val cmp = comparison(midVal)
        if (cmp < 0) low = mid + 1
        else if (cmp > 0) high = mid - 1
        else return mid // key found
    }
    return -(low + 1)  // key not found
}
```

The list is expected to be sorted so that the signs of the comparison function's return values ascend on the list elements, i.e. negative values come before zero and zeroes come before positive values. Otherwise, the result is undefined. If the list contains multiple elements for which comparison returns zero, there is no guarantee which one will be found.

```kotlin
@sample samples.collections.Collections.Lists.binarySearchWithComparator

List<T>().binarySearchWithComparator(fromIndex: Int = 0, toIndex: Int = size): Int =
    binarySearch(withIndex(fromIndex, toIndex), { fromIndex, toIndex ->
        comparator.compare(fromIndex, toIndex)
    })
```

The function returns zero when called on the list element being searched. On the elements coming before the target element, the function must return negative values. On the elements coming after the target element, the function must return positive values. If it is contained in the list within the specified range, otherwise, the inverted insertion point `(insertion point - 1)`.

```kotlin
@sample samples.collections.Collections.Lists.binarySearchWithComparisonFunction

List<T>().binarySearchWithComparisonFunction(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int =
    binarySearch(withIndex(fromIndex, toIndex), { fromIndex, toIndex ->
        comparison(fromIndex, toIndex)
    })
```

Checks that `from` and `to` are in the range of `0..size` and throws an appropriate exception, if they aren't. In private fun rangeCheck(size: Int, fromIndex: Int, toIndex: Int) { `\n        \n    }
```

IllegalStateException("fromIndex ($fromIndex) is greater than toIndex ($toIndex).")
```

throw IndexOutOfBoundsException("fromIndex ($fromIndex) is less than zero.
```

throw IndexOutOfBoundsException("fromIndex ($fromIndex) is greater than size ($size).
```

throw ArithmeticException("Index overflow has happened.
```

throw ArithmeticException("Count overflow has happened.
```

EmptyMap: Map<Any?, Nothing>, Serializable

package kotlin.collections

import kotlin.contracts.*

private object EmptyMap : Map<Any?, Nothing>, Serializable {
    private const val serialVersionUID: Long = 8246714829545688274

    override fun equals(other: Any?): Boolean = other is Map<*, *> &&
    other.isEmpty() || override fun hashCode(): Int = 0 ||
    override fun containsKey(key: Any?): Boolean = false

    override fun containsValue(value: Nothing): Boolean = false

    override fun get(key: Any?): Nothing? = null

    override fun entries: Set<Map.Entry<Any?, Nothing>> get() = EmptySet

    override fun keys: Set<Any?> get() = EmptySet

    override fun values: Collection<Nothing> get() = EmptyList

    private fun readResolve(): Any = EmptyMap
}

/**
 * Returns an empty read-only map of specified type.
 * @sample sample.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> emptyMap(): Map<K, V> = @Suppress("UNCHECKED_CAST") (EmptyMap as Map<K, V>)

/**
 * Returns a new read-only map with the specified contents, given as a list of pairs.
 * @sample sample.collections.Maps.Instantiation.mapFromPairs
 */
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =
    if (pairs.size > 0)
        pairs.toMap(LinkedHashMap(mapCapacity(pairs.size))) else emptyMap()

/**
 * Returns an empty new [MutableMap].
 * @sample sample.collections.Maps.Instantiation.emptyMutableMap
 */
@SinceKotlin("1.1")
public inline fun <K, V> mutableMapOf(): MutableMap<K, V> = LinkedHashMap()

/**
 * Returns a new [MutableMap] with the specified contents, given as a list of pairs.
 * @sample sample.collections.Maps.Instantiation.mutableMapFromPairs
 */
public fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns an empty new [HashMap].
 * @sample sample.collections.Maps.Instantiation.emptyHashMap
 */
@SinceKotlin("1.1")
public inline fun <K, V> hashMapOf(): HashMap<K, V> = HashMap<K, V>()

/**
 * Returns a new [HashMap] with the specified contents, given as a list of pairs.
 * @sample sample.collections.Maps.Instantiation.hashMapFromPairs
 */
public fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> =
    HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction]
 * and returning a read-only map with the same key-value pairs.
 * @sample sample.collections.Maps.Instantiation.linkedHashMapFromPairs
 */
@SinceKotlin("1.1")
public inline fun <K, V> linkedMapOf(): LinkedHashMap<K, V> = LinkedHashMap<K, V>()

/**
 * Returns a new [LinkedHashMap] with the specified contents, given as a list of pairs.
 * @sample sample.collections.Maps.Instantiation.linkedMapFromPairs
 */
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> =
    pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))
that function.

* Using it outside of the function produces an unspecified behavior.

* Entries of the map are iterated in the order they were added by the [builderAction].

* @sample samples.collections.Builders.Maps.buildMapSample

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline fun <K, V> buildMap(@BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
*     \n contract {
*         callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) \n*     } return buildMapInternal(builderAction)\n* }

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline fun <K, V> buildMap(capacity: Int, @BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
*     \n contract {
*         callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) \n*     } return buildMapInternal(capacity, builderAction)\n* }

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline fun <K, V> Map<K, V>?.isEmpty(): Boolean = this == null || isEmpty()\n* @sample samples.collections.Maps.Usage.mapIsNullOrEmpty

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline fun <K, V> Map<K, V>?.isNotEmpty(): Boolean = !isEmpty()\n* @sample samples.collections.Maps.Usage.mapIsNotEmpty

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline fun <K, V> Map<K, V>?.orEmpty(): Map<K, V> = this ?: emptyMap()\n
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
*     if (isEmpty()) defaultValue() else this\n
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.contains(key: K): Boolean = containsKey(key)\n* Returns the value corresponding to the given [key], or `null` if such a key is not present in the map.\n
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.get(key: K): V =@ Suppress("UNCHECKED_CAST") (this as Map<K, V>).get(key)\n* Returns `true` if the map contains the specified [key].\n
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.put(key: K, value: V): Unit {
*     put(key, value)\n* }

* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.containsKey(key: K): Boolean =null @ Suppress("UNCHECKED_CAST") (this as Map<K, V>).containsKey(key)\n* Returns `true` if the map maps one or more keys to the specified [value].\n
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public inline operator fun <@kotlin.internal.OnlyInputTypes K, V> Map<out K, V>.containsValue(value: V): Boolean =null @ Suppress("UNCHECKED_CAST") (this as Map<K, V>).containsValue(value)\n* Returns `true` if the map maps one or more keys to the specified [value].
Open Source Used In DNAC Platform Arica 7877

```kotlin
// false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
    this.containsKey(value) && this.containsValue(value)

@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>.containsKey(key: K): Boolean =
    this.containsKey(key)

@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.key(): K = key

@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.value(): V = value

@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = Pair(key, value)

@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.component1(): K = key

@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.component2(): V = value

/**
 * Removes the specified key and its corresponding value from this map.
 * @return the previous value associated with the key, or `null` if the key was not present in the map.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>.remove(key: K): V? =
    @Suppress("UNCHECKED_CAST") (this as Map<K, V>).remove(key)

/**
 * Returns the value for the given key, or the result of the [defaultValue] function if there was no entry for the given key.
 * @param key the key of the entry to retrieve
 * @param defaultValue the default value if no entry is found for the key
 * @return the value for the given key, or the default value
 */
@kotlin.internal.InlineOnly
@SinceKotlin("1.1")
public fun <K, V> Map<K, V>.getValue(key: K): V = getOrImplicitDefault(key)

/**
 * Returns a [MutableIterator] over the mutable entries in the [MutableMap].
 */
@kotlin.jvm.JvmName("mutableIterator")
@kotlin.internal.InlineOnly
    entries.iterator()

/**
 * Puts all the given [pairs] to this [MutableMap] with the first component in the pair being the key and the second the value.
 * Allows to overcome type-safety restriction of `remove` that requires to pass a key of type `K`.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> mapValuesTo(destination: V, transform: (Map.Entry<K, V>) -> V): V {
    return entries.associateByTo(destination, { it.key }, transform)
}

/**
 * Populates the given [destination] map with entries obtained
 * by applying the [transform] function to each entry in this map.
 * in case if any two entries are mapped to the equal keys, the value of the latter one will overwrite the value associated with the former one.
 * Allows to use destructuring declarations when working with maps, for example:
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>.mapTo(destination: V, transform: (Map.Entry<K, V>) -> V): V {
    return entries.associateByTo(destination, transform)
MutableMap\<K, in V>\.putAll(pairs: Array\<out Pair\<K, V>>): Unit \{ in put(key, value)\} \n \n\n** Puts all the elements of the given collection into this [MutableMap] with the first component in the pair being the key and the second the value.\n\n\nMutableMap\<K, in V>\.putAll(pairs: Iterable\<Pair\<K, V>>): Unit \{ in put(key, value)\} \n\n** Puts all the elements of the given sequence into this [MutableMap] with the first component in the pair being the key and the second the value.\n\n\nMutableMap\<K, in V>\.putAll(pairs: Sequence\<Pair\<K, V>>): Unit \{ in put(key, value)\} \n\n** Returns a new map with entries having the keys of this map and the values obtained by applying the [transform] function to each entry in this [Map].\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite the value with the former one.\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** In case if any two entries are mapped to the equal keys, the value of the latter one will overwrite the value with the former one.\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** Returns a map containing all key-value pairs with keys matching the given [predicate].\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** Returns a map containing all key-value pairs with keys matching the given [predicate].\n\n\n** The returned map preserves the entry iteration order of the original map.\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].\n\n\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].\n\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].\n\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].\n\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].\n\n\n** Returns a map containing all key-value pairs matching the given [predicate].
List) this[0] else iterator().next())
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
        }
    return toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()
}

/**
 * Populates and returns the 
 * mutable map with key-value pairs from the given collection of pairs.
 */
public fun <K, V, M : MutableMap<in K, in V>> Array<out Pair<K, V>>.toMap(): Map<K, V> = when (size) {
    0 -> emptyMap()
    1 -> mapOf(this[0])
    else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

/**
 * The returned map preserves the entry iteration order of the original array.
 * If any of two pairs would have the same key the last one gets added to the map.
 */
public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>())

/**
 * Returns a new map containing all key-value pairs from the given sequence of pairs.
 */
public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()

/**
 * Populates and returns the 
 * mutable map with key-value pairs from the given array of pairs.
 */
    destination.apply { putAll(this@toMap) }

/**
 * Creates a new read-only map by replacing or adding an entry to this map from a
 * given key-value [pair].
 */
    if (this.isEmpty()) mapOf(pair) else LinkedHashMap(this).apply { put(pair.first, pair.second) }

/**
 * Creates a new read-only map by replacing or adding entries to this map from a given
 * array of key-value [pairs].
 */
    if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }

/**
 * Creates a new read-only map by replacing or adding entries to this map from a given
 * sequence of key-value [pairs].
 */
    LinkedHashMap(this).apply { putAll(pairs) }

/**
 * Returns a new read-only map by replacing or adding entries to this map from another 
 * [map].
 */
    LinkedHashMap(this).apply { putAll(map) }

/**
 * Appends or replaces
 * the given [pair] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pair: Pair<K, V>) {
    put(pair.first, pair.second)
}
replaces all pairs from the given collection of [pairs] in this mutable map.

```kotlin
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Appends or replaces all pairs from the given array of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Array<out Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Appends or replaces all entries from the given [map] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(map: Map<K, V>) {
    putAll(map)
}

/**
 * Returns a map containing all entries of the original map except the entry with the given [key].
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(key) }.optimizeReadOnlyMap()

/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] collection.
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()

/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] array.
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()

/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] sequence.
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }.optimizeReadOnlyMap()

// do not expose for now
@PublishedApi
internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}
```

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SetsKt")
@file:OptIn(kotlin.experimental.ExperimentalTypeInference::class)

package kotlin.collections

import kotlin.contracts.*
empty read-only set. The returned set is serializable (JVM).

```kotlin
/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable (JVM).
 */
@sample samples.collections.Collections.Sets.emptyReadOnlySet
*public fun <T> emptySet(): Set<T> =
    EmptySet

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable (JVM).
 */
@sample samples.collections.Collections.Sets.readOnlySet
*public fun <T> setOf(vararg elements: T): Set<T> = if
    (elements.size > 0) elements.toSet() else emptySet()

/**
 * Returns an empty read-only set. The returned set is serializable (JVM).
 */
*public fun <T> emptySetOf(): Set<T> = emptySet()

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable (JVM).
 */
*public fun <T> setOfNot Null(element: T?): Set<T> = if
    (element != null) setOf(element) else emptySet()

/**
 * Returns a new read-only set with the given elements. Elements of the set are
 * iterated in the order they were specified.
 * The returned set is serializable (JVM).
 */
*public fun <T> setOfNotNull(vararg elements: T?): Set<T> {
    return elements.filterNotNullTo(LinkedHashSet())
}

/**
 * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]
 * and returning a read-only set with the same elements.
 * The set passed as a receiver to the [builderAction] is valid only inside that function.
 * Using it outside of the function produces an unspecified behavior. Elements of the set are
 * iterated in the order they were added by the [builderAction].
 */
*public fun <E> buildSet(@BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> =
    buildSetInternal(builderAction)
```

```kotlin
@SinceKotlin("1.1")@kotlin.internal.InlineOnly
public inline fun <T> mutableSetOf(): Set<T> =
    elements.toCollection(LinkedHashSet())

/**
 * Returns a new [MutableSet] with the given elements. Elements of the set are
 * iterated in the order they were specified.
 */
public fun <T> mutableSetOf(vararg elements: T): Set<T> =
    elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))

/**
 * Returns an empty new [HashSet].
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> hashSetOf(): Set<T> =
    HashSet()

/**
 * Returns a new [HashSet] with the given elements. Elements of the set are
 * iterated in the order they were specified.
 */
public fun <T> hashSetOf(vararg elements: T): Set<T> =
    elements.toCollection(HashSet(mapCapacity(elements.size)))

/**
 * Returns an empty new [LinkedHashSet].
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> linkedSetOf(): LinkedHashSet<T> =
    LinkedHashSet()

/**
 * Returns a new [LinkedHashSet] with the given elements. Elements of the set are
 * iterated in the order they were specified.
 */
public fun <T> linkedSetOf(vararg elements: T): LinkedHashSet<T> =
    elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))

```
buildSet(capacity: Int, @BuilderInference builderAction: MutableSet<@BuilderInference<T>>.() -> Unit): Set<T> { internal val contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) } return buildSetInternal(capacity, builderAction) }

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal expect inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E>

/** Returns this Set if it's not `null` and the empty set otherwise. */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.orEmpty(): Set<T> = this ?: emptySet()

internal fun <T> Set<T>.optimizeReadOnlySet() = when (size) {
    0 -> emptySet()
    1 -> setOf(iterator().next())
    else -> this
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")
@file:Suppress("PLATFORM_CLASS_MAPPED_TO_KOTLIN")

package kotlin.text

/** Parses the string as a signed [Byte] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)

/** Parses the string as a signed [Byte] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toByteOrNull(radix: Int): Byte? {
    val int = this.toIntOrNull(radix) ?: return null
    if (int < Byte.MIN_VALUE || int > Byte.MAX_VALUE) return null
    return int.toByte()
}

/** Parses the string as a [Short] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toShortOrNull(): Short? = toShortOrNull(radix = 10)

/** Parses the string as a [Short] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toShortOrNull(radix: Int): Short? {
    val int = this.toIntOrNull(radix) ?: return null
    if (int < Short.MIN_VALUE || int > Short.MAX_VALUE) return null
    return int.toShort()
}

/** Parses the string as an [Int] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toIntOrNull(): Int? = toIntOrNull(radix = 10)

/** Parses the string as an [Int] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toIntOrNull(radix: Int): Int? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Int

    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after

        start = 1

        if (firstChar == '‐') {
            isNegative = true
            limit = Int.MIN_VALUE
        } else if (firstChar == '‐') {
            isNegative = false
            limit = -Int.MAX_VALUE
        } else if (firstChar == '+) {
            isNegative = false
            limit = -Int.MAX_VALUE
        } else return null
    } else if (firstChar == '0') {
        isNegative = false
        limit = -Int.MAX_VALUE
    } else else return null

    else if (start == 0) isNegative = false
    limit = -Int.MAX_VALUE

    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result < limit) result += digit
    }

    return if (isNegative) result else -result
}

/** Parses the string as a [Long] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toLongOrNull(): Long? = toLongOrNull(radix = 10)

/** Parses the string as a [Long] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Long

    val firstChar = this[0]
    if (firstChar == '‐') {
        isNegative = true
        limit = Int.MIN_VALUE
    } else if (firstChar == '+) {
        isNegative = false
        limit = -Int.MAX_VALUE
    } else else return null

    else if (start == 0) isNegative = false
    limit = -Int.MAX_VALUE

    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limit) result += digit
    }

    return if (isNegative) result else -result
}

/** Parses the string as a [Double] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toDoubleOrNull(): Double? = toDoubleOrNull(radix = 10)

/** Parses the string as a [Double] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toDoubleOrNull(radix: Int): Double? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Double

    val firstChar = this[0]
    if (firstChar == '‐') {
        isNegative = true
        limit = Int.MIN_VALUE
    } else if (firstChar == '+) {
        isNegative = false
        limit = -Int.MAX_VALUE
    } else else return null

    else if (start == 0) isNegative = false
    limit = -Int.MAX_VALUE

    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limit) result += digit
    }

    return if (isNegative) result else -result
}

/** Parses the string as a [Float] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toFloatOrNull(): Float? = toFloatOrNull(radix = 10)

/** Parses the string as a [Float] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toFloatOrNull(radix: Int): Float? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Float

    val firstChar = this[0]
    if (firstChar == '‐') {
        isNegative = true
        limit = Int.MIN_VALUE
    } else if (firstChar == '+) {
        isNegative = false
        limit = -Int.MAX_VALUE
    } else else return null

    else if (start == 0) isNegative = false
    limit = -Int.MAX_VALUE

    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limit) result += digit
    }

    return if (isNegative) result else -result
}

/** Parses the string as a [Complex] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toComplexOrNull(): Complex? = toComplexOrNull(radix = 10)

/** Parses the string as a [Complex] number and returns the result or `null` if the string is not a valid representation of a number. */
@SinceKotlin("1.1")
public fun String.toComplexOrNull(radix: Int): Complex? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Complex

    val firstChar = this[0]
    if (firstChar == '‐') {
        isNegative = true
        limit = Int.MIN_VALUE
    } else if (firstChar == '+) {
        isNegative = false
        limit = -Int.MAX_VALUE
    } else else return null

    else if (start == 0) isNegative = false
    limit = -Int.MAX_VALUE

    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limit) result += digit
    }

    return if (isNegative) result else -result
}
null isNegative = true
    limit = Long.MIN_VALUE
} else if (firstChar == '+') {
    isNegative = false
    limit = -Long.MAX_VALUE
} else
    return null
}

var limitBeforeMul = limitForMaxRadix

val limitBeforeMul = limit / radix\n\nif (result < limitBeforeMul) {
    return result
} else

val limitForMaxRadix = (-Long.MAX_VALUE) / 36

var limitBeforeMul = limitForMaxRadix

\n\nif (isNegative) result else -result
}

numberFormatError(input: String): Nothing = throw
NumberFormatException("Invalid number format: '$input'")

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmName("UnsignedKt")
@file:OptIn(ExperimentalUnsignedTypes::class)
npackage kotlin

@PublishedApi
internal fun uintCompare(v1: Int, v2: Int): Int = (v1 xor Int.MIN_VALUE).compareTo(v2 xor Int.MIN_VALUE)

@PublishedApi

@PublishedApi
internal fun uintDivide(v1: UInt, v2: UInt): UInt = (v1.toLong() / v2.toLong()).toUInt()

@PublishedApi
internal fun uintRemainder(v1: UInt, v2: UInt): UInt = (v1.toLong() % v2.toLong()).toUInt()

@PublishedApi
internal fun ulongDivide(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63:
        return if (v1 < v2) ULong(0) else ULong(1)
    } else

    // Optimization - use signed division if both dividend and divisor < 2^63
    if (dividend < 0) {
        return ULong(dividend / divisor)
    } else // Otherwise, approximate the quotient, check, and correct if necessary.
        return ULong(dividend / divisor).toInt() shl 1

    // Sometimes, dividend >= divisor * 2^63
    if (rem >= divisor) {
        rem -= divisor
        quotient += 1
    }

    return quotient.toString(10) + rem.toString(10)
}

@PublishedApi
internal fun ulongToString(v: Long, base: Int): String {
    if (v >= 0) return v.toString(base)

    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient * base

    if (rem >= base)
        rem -= base

    return quotient.toString(base) + rem.toString(base)
}

ulongToString(v: Long): String = ulongToString(v, 10)

ulongToString(v: Long, base: Int): String {
    if (v >= 0) return v.toString(base)

    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient * base

    if (rem >= base)
        rem -= base

    return quotient.toString(base) + rem.toString(base)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmName("UnsignedKt")
@file:OptIn(ExperimentalUnsignedTypes::class)
npackage kotlin

@PublishedApi
internal fun uintCompare(v1: Int, v2: Int): Int = (v1 xor Int.MIN_VALUE).compareTo(v2 xor Int.MIN_VALUE)

@PublishedApi

@PublishedApi
internal fun uintDivide(v1: UInt, v2: UInt): UInt = (v1.toLong() / v2.toLong()).toUInt()

@PublishedApi
internal fun uintRemainder(v1: UInt, v2: UInt): UInt = (v1.toLong() % v2.toLong()).toUInt()

@PublishedApi
internal fun ulongDivide(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63:
        return if (v1 < v2) ULong(0) else ULong(1)
    }

    // Optimization - use signed division if both dividend and divisor < 2^63
    if (dividend < 0) {
        return ULong(dividend / divisor)
    } else // Otherwise, approximate the quotient, check, and correct if necessary.
        return ULong(dividend / divisor).toInt() shl 1

    // Sometimes, dividend >= divisor * 2^63
    if (rem >= divisor) {
        rem -= divisor
        quotient += 1
    }

    return quotient.toString(10) + rem.toString(10)
}

@PublishedApi
internal fun ulongToString(v: Long, base: Int): String {
    if (v >= 0) return v.toString(base)

    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient * base

    if (rem >= base)
        rem -= base

    return quotient.toString(base) + rem.toString(base)
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("$CollectionsKt")

package kotlin.collections

/**
 * Given an [iterator] function constructs an [Iterable] instance that returns values through the [Iterator] provided by that function. * @sample samples.collections.Iterables.Building.iterable
 * Overide fun iterator(): [Iterator]<T> = object { override fun iterator(): Iterator<T> = Iterator() }
 */

public fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
    override fun iterator(): Iterator<T> = iterator()
}

/**
 * A wrapper over another [Iterable] or any other object that can produce an [Iterator] that returns * an indexing iterator. */

@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrNull(): Int? = when (this) {
    is Collection<T> -> this.size
    else -> null
}

/**
 * Returns the size of this iterable if it is known, or `null` otherwise. */

@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrDefault(default: Int): Int = when (this) {
    is Collection<T> -> this.size
    else -> default
}

/**
 * Converts this collection to a set, when it's safe to convert this collection to a set without changing contains method behavior. */

private fun <T> Collection<T>.safeToConvertToSet() = this is ArrayList

/** Converts this collection to a set, when it's worth so and it doesn't change contains method behavior. */

internal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> = when (this) {
    is Set -> this
    else -> if (this.safeToConvertToSet()) toHashSet() else this
}

/**
 * Returns a single list of all elements from all collections in the given collection. * @sample samples.collections.Iterables.Operations.flattenIterable
 */

public fun <T> Iterable<Iterable<T>>.flatten(): List<T> {
    val result = ArrayList<T>()
    for (element in this) {
        result.addAll(element)
    }
    return result
}

/**
 * Returns a pair of lists, where
 * *first* list is built from the first values of each pair from this collection,
 * *second* list is built from the second values of each pair from this collection. * @sample samples.collections.Iterables.Operations.unzipIterable
 */

public fun <T, R> Iterable<Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {
    val expectedSize = collectionSizeOrDefault(10)
    val listT = ArrayList<T>(expectedSize)
    val listR = ArrayList<R>(expectedSize)
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

Open Source Used In DNAC Platform Arica  7884
drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
    override fun drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
}

/**
 * Returns a sequence that iterates through the elements either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.
 */@sample
open fun DropSequence<T>?: DropSequence<T> = this ?: emptySequence()

/**
 * Returns a sequence that iterates through the elements either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.
 * @sample
 * @sinceKotlin("1.3")
 * @kotlin.internal.InlineOnly
 */@sample
open fun DropSequence<T>?: DropSequence<T> = this ?: emptySequence()

/**
 * Returns a sequence that iterates through the elements either of this sequence or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.
 */@sample
open fun DropSequence<T>?: DropSequence<T> = this ?: emptySequence()
sequence: Sequence<T>, private val transformer: (T) -> R) : Sequence<R> {
    override fun iterator(): Iterator<R>
        = object : Iterator<R> {
        val iterator = sequence.iterator()
        override fun next(): R {
            return

        }

        override fun hasNext(): Boolean {
            return iterator.hasNext()
        }
    }

    internal fun <E> flatten(iterator: (R) -> Iterator<E>): Sequence<E> {
        return FlatteningSequence<T, R, E>(sequence, transformer, iterator)
    }
}

/**
 * A sequence which returns the 
 * results of applying the given [transformer] function to the values
 * in the underlying [sequence], where the transformer function takes
 * the index of the value in the underlying n * in the [sequence].
 */
internal class TransformingIndexedSequence<T, R> {
    constructor(private val sequence: Sequence<T>, private val transformer: (Int, T) -> R) : Sequence<R> {
        override fun iterator(): Iterator<R> = object : Iterator<R> {

            override fun next(): R {
                return
            }

            override fun hasNext(): Boolean {
                return iterator.hasNext()
            }
        }
    }
}

/**
 * A sequence which combines values from the underlying
 * [sequence] with their indices and returns them as
 * [IndexedValue] objects.
 */
internal class IndexingSequence<T> {
    constructor(private val sequence: Sequence<T>) : Sequence<IndexedValue<T>> {
        override fun iterator(): Iterator<IndexedValue<T>> = object : Iterator<IndexedValue<T>> {

            override fun next(): IndexedValue<T> {
                return
            }

            override fun hasNext(): Boolean {
                return iterator.hasNext()
            }
        }
    }
}

/**
 * A sequence that supports drop(n) and take(n) operations
 */
internal interface DropTakeSequence<T> : Sequence<T> {
    fun drop(n: Int): Sequence<T>
    fun take(n: Int): Sequence<T>
}

/**
 * A sequence that skips [startIndex] values from the underlying [sequence] and
 * stops returning values right before [endIndex], i.e. stops at endIndex - 1
 */
internal class SubSequence<T>(
    private val sequence: Sequence<T>,
    private val startIndex: Int,
    private val endIndex: Int
) : Sequence<T>,
DropTakeSequence<T> {

    init {
        require(startIndex >= 0) { "startIndex should be non-negative, but is $startIndex" }
        require(endIndex >= 0) { "endIndex should be non-negative, but is $endIndex" }
        require(endIndex >= startIndex) { "endIndex should be not less than startIndex, but was $endIndex < $startIndex" }
    }

    private val count: Int get() = endIndex - startIndex

    override fun drop(n: Int): Sequence<T> = if (n 
        >= count) emptySequence() else SubSequence(sequence, startIndex + n, endIndex)
    override fun take(n: Int): 
@Suppress("UNCHECKED_CAST")
val result = nextItem as T
nextItem = null

override fun hasNext(): Boolean {
    if (dropState == -1)
        drop()
    return dropState == 1 ||
    iterator.hasNext()
}

override fun override func hasNext(): Boolean {
    if (dropState == -1)
        drop()
    return dropState == 1 ||
    iterator.hasNext()
}

private class GeneratorSequence<T : Any>(private val getInitialValue: () -> T?, private val getNextValue: (T) -> T?) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        var nextItem: T? = null
        var nextState: Int = -2 // -2 for initial unknown, -1 for next unknown, 0 for done, 1 for continue
        var nextItem: T? = null

        private fun calcNext() {
            nextItem = if (nextState == -2) getInitialValue() else getNextValue(nextItem!!)
            nextState = if (nextItem == null) 0 else
            val next = source.next()
            val key = keySelector(next)
            if (observed.add(key)) {
                setNext(next)
                return
            }
        }

        override fun next(): T {
            if (nextState < 0)
                calcNext()
            if (nextState == 0)
                throw NoSuchElementException()
            val result = nextItem as T
            // Do not clean
            nextItem (to avoid keeping reference on yielded instance) -- need to keep state for getNextValue
            nextState = -1
            return result
        }

        override fun hasNext(): Boolean {
            if (nextState < 0)
                calcNext()
            return nextState == 1
        }
    }
}

/**
 * Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value,
 * and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration.
 * The sequence can be iterated multiple times.
 * @see kotlin.collections.Sequences.build
 */
public fun <T : Any> generateSequence(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T> = GeneratorSequence(seedFunction, nextFunction)

/**
 * Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value, and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration. The sequence can be iterated multiple times.
 * @see kotlin.collections.Sequences.build
 */
public fun <T : Any> generateSequenceWithSeed(seed: T?, nextFunction: (T) -> T?): Sequence<T> = GeneratorSequence(seed, nextFunction)

/**
 * Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value, and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration. The sequence can be iterated multiple times.
 * @see kotlin.collections.Sequences.build
 */
public fun <T : Any> generateSequenceWithLazySeed(): Sequence<T> = GeneratorSequence(seedFunction = () -> T?, nextFunction = (T) -> T?)

/**
 * Returns a sequence defined by the function [seedFunction], which is invoked to produce the starting value, and the [nextFunction], which is invoked to calculate the next value based on the previous one on each iteration. The sequence can be iterated multiple times.
 * @see kotlin.collections.Sequences.build
 */
public fun <T : Any> generateSequenceWithSeedFunction(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T> = GeneratorSequence(seedFunction, nextFunction)

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("\nimport kotlin.contracts.

Open Source Used In DNAC Platform Arica 7888
*\n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.*\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.*\n * * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.*\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.*\n
Open Source Used In DNAC Platform Arica
other types are compared for equality with the \([\text{equals}]\)[Any.equals] function.

For floating point numbers it means that `'NaN'` is equal to itself and `-0.0` is not equal to `0.0`.

*\@SinceKotlin("1.1\")*\@kotlin.internal.LowPriorityInOverloadResolution
public actual infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean {
    return this.contentDeepEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *deeply* equal to one another, i.e. contain the same number of the same elements in the same order.

The specified arrays are also considered deeply equal if both are `null`. If two corresponding elements are nested arrays, they are also compared deeply. If any of arrays contains itself on any nesting level the behavior is undefined.

The elements of other types are compared for equality with the \([\text{equals}]\)[Any.equals] function.

For floating point numbers it means that `'NaN'` is equal to itself and `-0.0` is not equal to `0.0`.

*\@SinceKotlin("1.1\")*\@kotlin.internal.LowPriorityInOverloadResolution
public actual infix fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is \([\text{List}]\). Nested arrays are treated as lists too.

If any of arrays contains itself on any nesting level the behavior is undefined.

*\@SinceKotlin("1.1\")*\@library("arrayDeepHashCode")
public actual fun <T> Array<out T>?.contentDeepHashCode(): Int {
    definedExternally
}

/**
 * Returns a string representation of the contents of this array as if it is \([\text{List}]\).

Nested arrays are treated as lists too.

If any of arrays contains itself on any nesting level the behavior is undefined.

*\@SinceKotlin("1.1\")*\@library("arrayDeepToString")
public actual fun <T> Array<out T>.contentDeepToString(): String {
    return this.contentDeepToString()
}

/**
 * Returns a string representation of the contents of this array as if it is \([\text{List}]\).

Nested arrays are treated as lists too.

If any of arrays contains itself on any nesting level the behavior is undefined.

*\@SinceKotlin("1.1\")*\@library("arrayDeepToString")
public actual fun <T> Array<out T>?.contentDeepToString(): String {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,

Instructed equal to one another, i.e. contain the same number of the same elements in the same order.

The elements are compared for equality with the \([\text{equals}]\)[Any.equals] function.

For floating point numbers it means that `'NaN'` is equal to itself and `-0.0` is not equal to `0.0`.

*\@SinceKotlin("1.1\")*\@library("arrayDeepEquals")
public actual infix fun <T> Array<out T>? contentDeepEquals(other: Array<out T>): Boolean {
    return this.contentDeepEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,

Instructed equal to one another, i.e. contain the same number of the same elements in the same order.

The elements are compared for equality with the \([\text{equals}]\)[Any.equals] function.

For floating point numbers it means that `'NaN'` is equal to itself and `-0.0` is not equal to `0.0`.

*\@SinceKotlin("1.1\")*\@library("arrayDeepEquals")
public actual infix fun <T> ByteArray.contentDeepEquals(other: ByteArray): Boolean {
    return this.contentDeepEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,

Instructed equal to one another, i.e. contain the same number of the same elements in the same order.

The elements are compared for equality with the \([\text{equals}]\)[Any.equals] function.

For floating point numbers it means that `'NaN'` is equal to itself and `-0.0` is not equal to `0.0`.

*\@SinceKotlin("1.1\")*\@library("arrayDeepEquals")
public actual infix fun <T> ShortArray.contentDeepEquals(other: ShortArray): Boolean {
    return this.contentDeepEquals(other)
}
* The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun IntArray.contentEquals(other: IntArray): Boolean {
    return this.contentEquals(other)
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun LongArray.contentEquals(other: LongArray): Boolean {
    return this.contentEquals(other)
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun FloatArray.contentEquals(other: FloatArray): Boolean {
    return this.contentEquals(other)
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean {
    return this.contentEquals(other)
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean {
    definedExternally
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {
    definedExternally
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean {
    definedExternally
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {
    definedExternally
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun <T> Array<out T>?.contentEquals(other: Array<out T>?): Boolean {
    definedExternally
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.  

```kotlin
@SinceKotlin("1.4")
@library("arrayEquals")
public actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {
    definedExternally
}
```
fun contentEquals(other: ShortArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: IntArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: LongArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: FloatArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: DoubleArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: BooleanArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentEquals(other: CharArray?): Boolean {
    definedExternally
    * Returns `true` if the two specified arrays are
    * structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
    * The elements are compared for equality with the `equals` function.
    * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
}

fun contentHashCode(): Int {
    return this.contentHashCode()
warning.

@DeprecatedSinceKotlin("1.1")
public actual fun FloatArray.contentHashCode(): Int { 
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
@DeprecatedSinceKotlin("1.4")
public actual fun DoubleArray.contentHashCode(): Int { 
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
@DeprecatedSinceKotlin("1.4")
public actual fun BooleanArray.contentHashCode(): Int { 
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
@DeprecatedSinceKotlin("1.4")
public actual fun CharArray.contentHashCode(): Int { 
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
@DeprecatedSinceKotlin("1.4")
public actual fun <T> Array<out T>?.contentHashCode(): Int { 
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual fun FloatArray.contentToString(): String { 
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual fun DoubleArray.contentToString(): String { 
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual fun BooleanArray.contentToString(): String { 
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual fun CharArray.contentToString(): String { 
    return this.contentToString()
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.")
@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual fun <T> Array<out T>?.contentToString(): String { 
    definedExternally
}
```kotlin
actual fun LongArray.contentToString(): String { return this.contentToString() } /* Get a string representation of the contents of the specified array as if it is [List]. */

fun @Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.") @SinceKotlin("1.4") @library("arrayToString") public actual fun Array<out T>?.contentToString(): String {
    definedExternally
}

fun @Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.") @SinceKotlin("1.4") @library("arrayToString") public actual fun ByteArray?.contentToString(): String {
    definedExternally
}

fun @Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.") @SinceKotlin("1.4") @library("arrayToString") public actual fun CharArray?.contentToString(): String {
    definedExternally
}
```

---

Open Source Used In DNAC Platform Arica  7895
the subrange to copy, 0 by default.\n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n * @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * @return the [destination] array.\n
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {\n    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)\n    return destination}\n
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this.unsafeCast<Array<Byte>>(), destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)
    return destination}\n
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray {
    arrayCopy(this.unsafeCast<Array<Short>>(), destination.unsafeCast<Array<Short>>(), destinationOffset, startIndex, endIndex)
    return destination}\n
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray {
    arrayCopy(this.unsafeCast<Array<Int>>(), destination.unsafeCast<Array<Int>>(), destinationOffset, startIndex, endIndex)
    return destination}
Copies this array or its subrange into the [destination] array and returns that array.

- If it's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
- @param destination the array to copy to.
- @param destinationOffset the position in the [destination] array to copy to, 0 by default.
- @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
- @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray {
    arrayCopy(this.unsafeCast<Array<Long>>(), destination.unsafeCast<Array<Long>>(), destinationOffset, startIndex, endIndex)
    return destination
}

Copies this array or its subrange into the [destination] array and returns that array.

- @param destination the array to copy to.
- @param destinationOffset the position in the [destination] array to copy to, 0 by default.
- @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
- @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray {
    arrayCopy(this.unsafeCast<Array<Float>>(), destination.unsafeCast<Array<Float>>(), destinationOffset, startIndex, endIndex)
    return destination
}

Copies this array or its subrange into the [destination] array and returns that array.

- @param destination the array to copy to.
- @param destinationOffset the position in the [destination] array to copy to, 0 by default.
- @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
- @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray {
    arrayCopy(this.unsafeCast<Array<Double>>(), destination.unsafeCast<Array<Double>>(), destinationOffset, startIndex, endIndex)
    return destination
}
out of the [destination] array indices range.\n* \n* @return the [destination] array.\n*\n*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray \n{\narrayCopy(this.unsafeCast<Array<Boolean>>(), destination.unsafeCast<Array<Boolean>>(), destinationOffset, startIndex, endIndex)\n    return destination[un]??\n* Copies this array or its subrange into the [destination] array and returns that array.\n* \n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* \n* @param destination the array to copy to.\n* \n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* \n* @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* \n* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.\n* \n* @return the [destination] array.\n*\n*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray \n{\narrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)\n    return destination[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyOf\n*/\npublic actual inline fun <T> Array<out T>.copyOf(): Array<T> \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun ByteArray.copyOf(): ByteArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun ShortArray.copyOf(): ShortArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun IntArray.copyOf(): IntArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun LongArray.copyOf(): LongArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun BooleanArray.copyOf(): BooleanArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/\npublic actual inline fun CharArray.copyOf(): CharArray \n{\n    return this.asDynamic().slice[un]??\n* Returns new array which is a copy of the original array.\n* \n* \n* @sample samples.collections.Arrays.CopyOfOperations.copyWith\n*/

samples.collections.Arrays.copyOfOperations.resizedPrimitiveCopyOf

*anpublic actual fun
ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

*anpublic actual fun
ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))
}

* Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
return withType("CharArray", fillFrom(this, CharArray(newSize)))

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with `null` values if necessary.
 * If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun <T> Array<out T>.copyOf(newSize: Int):
    return withType("CharArray", fillFrom(this, CharArray(newSize)))

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun <T> Array<out T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is truncated to the [toIndex].
 * - If [fromIndex] is greater than [toIndex], the copy array is truncated to the [toIndex].
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is truncated to the [toIndex].
 * - If [fromIndex] is greater than [toIndex], the copy array is truncated to the [toIndex].
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is truncated to the [toIndex].
 * - If [fromIndex] is greater than [toIndex], the copy array is truncated to the [toIndex].
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is truncated to the [toIndex].
 * - If [fromIndex] is greater than [toIndex], the copy array is truncated to the [toIndex].
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is truncated to the [toIndex].
 * - If [fromIndex] is greater than [toIndex], the copy array is truncated to the [toIndex].
 * - If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, the copy array is filled with `null` values.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}
fromIndex the start of the range (inclusive) to fill, 0 by default.

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\@SinceKotlin("1.3")\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\*/
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * 
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\@SinceKotlin("1.3")\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\*/
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * 
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\@SinceKotlin("1.3")\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\*/
public actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * 
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\@SinceKotlin("1.3")\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\*/
public actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*\@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")\*/
public actual inline operator fun <T> Array<out T>.plus(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*\@Suppress("NOTHING_TO_INLINE")\*/
public actual inline operator fun ByteArray.plus(element: Byte): ByteArray {
    return this plus byteArrayOf(element)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*\@Suppress("NOTHING_TO_INLINE")\*/
public actual inline operator fun ShortArray.plus(element: Short): ShortArray {
    return this plus shortArrayOf(element)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*/
public actual inline operator fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*/
public actual inline operator fun <T> Array<out T>.plus(element: T): Array<T> {
    return this asDynamic().concat(arrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*/
public actual inline operator fun ByteArray.plus(element: Byte): ByteArray {
    return plus byteArrayOf(element)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

*/
public actual inline operator fun ShortArray.plus(element: Short): ShortArray {
    return plus shortArrayOf(element)
}
original array and then the given [element].

```kotlin
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun IntArray.plus(element: Int): IntArray {
    return plus(intArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun LongArray.plus(element: Long): LongArray {
    return plus(longArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun FloatArray.plus(element: Float): FloatArray {
    return plus(floatArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun DoubleArray.plus(element: Double): DoubleArray {
    return plus(doubleArrayOf(element))
}
```

```kotlin
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun BooleanArray.plus(element: Boolean): BooleanArray {
    return plus(booleanArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT," "NOTHING_TO_INLINE")
public actual operator fun <T> Array<out T>.plus(elements: Collection<T>): Array<T> {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun ShortArray.plus(elements: Collection<Short>): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun IntArray.plus(elements: Collection<Int>): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun LongArray.plus(elements: Collection<Long>): LongArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
```

```kotlin
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
```

```kotlin
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
```

```kotlin
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
```
public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun IntArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun LongArray.sort(): Unit {
    if (size > 1) sort {
        a: Long, b: Long -> a.compareTo(b)
    }
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun ByteArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun ShortArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun FloatArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun CharArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place according to the natural order of its elements.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sort() {
    if (size > 1) sortArray(this)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sort(comparison: (a: T, b: T) -> Int) {
    if (size > 1) sortArrayWith(this, comparison)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexed() {
    if (size > 1) sortArrayIndexedWith(this)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexedWith() {
    if (size > 1) sortArrayIndexedWith(this)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexedWith(vararg args: Int) {
    if (size > 1) sortArrayIndexedWith(this, *args)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexedWith(vararg args: Int, @param fromIndex: Int) {
    if (size > 1) sortArrayIndexedWith(this, *args, fromIndex)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexedWith(vararg args: Int, @param fromIndex: Int) {
    if (size > 1) sortArrayIndexedWith(this, *args, fromIndex)
}

/**
 * Sorts a range in the array in-place.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 */
public actual fun <T : Comparable<T>> Array<out T>.sortIndexedWith(vararg args: Int, @param fromIndex: Int) {
    if (size > 1) sortArrayIndexedWith(this, *args, fromIndex)
}
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    sortArrayWith(this, fromIndex, toIndex, naturalOrder())\n} \n\n*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ByteArray>()\n    subarray.sort()\n} \n\n*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ShortArray>()\n    subarray.sort()\n} \n\n*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<IntArray>()\n    subarray.sort()\n} \n\n*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    sortArrayWith(this.unsafeCast<Array<Long>>(), fromIndex, toIndex, naturalOrder())\n} \n\n*\n@SinceKotlin("1.4")\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n\npublic actual fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    val subarray = this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<FloatArray>()\n    subarray.sort()\n}
**
 * Returns a *typed* object array containing all of the elements of this primitive array.

```kotlin
@file:kotlin.jvm.JvmName("ComparisonsKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.comparisons

/**
 * Compares two values using the specified functions [selectors] to calculate the result
 * of the comparison.
 */
public fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>?): Int {
    require(selectors.size > 0)
    return compareValuesByImpl(a, b, selectors)
}

private fun <T> compareValuesByImpl(a: T, b: T, selectors: Array<out (T) -> Comparable<*>?>): Int {
    for (fn in selectors) {
        val v1 = fn(a)
        val v2 = fn(b)
        val diff = compareValues(v1, v2)
        if (diff != 0) return diff
    }
    return 0
}

/**
 * Compares two values using the specified [selector] function to calculate the result of the comparison.
 */
public inline fun <T> compareValuesBy(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValues(selector(a), selector(b))
}

/**
 * Compares two values using the specified [selector] function to calculate a result of comparison.
 * The functions are called sequentially, receive the given values [a] and [b] and return
 * [Comparable] objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do
 * not compare as equal, the result of that comparison is returned.
 */
@kotlin.internal.InlineOnly
public inline fun <T, K> compareValuesBy(a: T, b: T, comparator: Comparator<in K>, selector: (T) -> K): Int {
    return comparator.compare(selector(a), selector(b))
}

/**
 * Creates a comparator using the sequence of functions to calculate a result of comparison.
 */
@Suppress("NOTHING_TO_INLINE")
public inline fun <T> compareValuesWith(a: T, b: T, comparator: Comparator<T>): Int {
    return comparator.compare(a, b)
}

/**
 * Compares two nullable [Comparable] values. Null is considered less than any value.
 */
public fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {
    if (a === b) return 0
    if (a == null) return -1
    if (b == null) return 1
    @Suppress("UNCHECKED_CAST")
    return (a as Comparable<Any>).compareTo(b)
}
```

---

**Open Source Used In DNAC Platform Arica  7907**
no

* compare as equal, the result of that comparison is returned from the [Comparator].

* @sample

samples.comparisons.Comparisons.compareByWithSelectors

*/

public fun <T> compareBy(vararg selectors: (T) -> Comparable<*?>): Comparator<T> {
    require(selectors.size > 0)
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}

/**
 * Creates a comparator using the function to transform value to a [Comparable] instance for comparison.

 * @sample

samples.comparisons.Comparisons.compareByWithSingleSelector

*/

public fun <T> compareBy(selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(a, b, selector)
    }

/**
 * Creates a comparator using the [selector] function to transform values being compared and then

 * applying * the specified [comparator] to compare transformed values.

 * @sample

samples.comparisons.Comparisons.compareByWithComparator

*/

public inline fun <T, K> compareBy(comparator: Comparator<in K>, selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(b, a, comparator, selector)
    }

/**
 * Creates a comparator comparing values after the primary comparator defined them equal. It

 * uses * the [selector] function to transform values and then compares them with the given [comparator].

 * @sample

samples.comparisons.Comparisons.thenBy

*/

public inline fun <T> Comparator<T>.thenBy(selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(a, b)
        if (previousCompare != 0) previousCompare else
            compareValuesBy(a, b, selector)
    }

/**
 * Creates a descending comparator comparing values after the primary comparator defined them equal. It

 * uses * the [selector] function to transform values and then compares them with the given [comparator].

 * @sample

samples.comparisons.Comparisons.thenByDescending

*/

public inline fun <T> Comparator<T>.thenByDescending(selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(b, a)
        if (previousCompare != 0) previousCompare else
            compareValuesBy(b, a, selector)
    }

/**
 * Creates a comparator using the primary comparator and function to calculate a result of comparison.

 * @sample

samples.comparisons.Comparisons.thenComparator

*/

public inline fun <T> Comparator<T>.thenComparator(comparison: (a: T, b: T) -> Int): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(a, b)
        if (previousCompare != 0) previousCompare else
            compareValuesBy(a, b, comparison)
    }

* /
Comparator { a, b ->
    val previousCompare = this@thenComparator.compare(a, b)
    if (previousCompare != 0) previousCompare
    else comparison(a, b)
}

/**
 * Combines this comparator and the given [comparator] such that the latter is applied only when the former considered values equal.
 */
public infix fun <T> Comparator<T>.then(comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@then.compare(a, b)
        if (previousCompare != 0) previousCompare
        else comparator.compare(a, b)
    }

/**
 * Combines this comparator and the given [comparator] such that the latter is applied only when the former considered values equal.
 */
public infix fun <T> Comparator<T>.thenDescending(comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenDescending.compare(a, b)
        if (previousCompare != 0) previousCompare
        else comparator.compare(b, a)
    }

// Not so useful without type inference for receiver of expression
/**
 * Extends the given [comparator] of non-nullable values to a comparator of nullable values considering `null` value less than any other value.
 */
public fun <T : Any> nullsFirst(comparator: Comparator<in T>): Comparator<T?> =
    Comparator { a, b ->
        when {
            a == null -> -1
            b == null -> 1
            else -> comparator.compare(a, b)
        }
    }

/**
 * Extends the given [comparator] of non-nullable values to a comparator of nullable values considering `null` value greater than any other value.
 */
public fun <T : Any> nullsLast(comparator: Comparator<in T>): Comparator<T?> =
    Comparator { a, b ->
        when {
            a == null -> 1
            b == null -> -1
            else -> comparator.compare(a, b)
        }
    }

/**
 * Returns a comparator that compares [Comparable] objects in natural order.
 */
public fun <T : Comparable<T>> naturalOrder(): Comparator<T> = @Suppress("UNCHECKED_CAST") (NaturalOrderComparator as Comparator<T>)

/**
 * Returns a comparator that compares [Comparable] objects in reversed natural order.
 */
public fun <T : Comparable<T>> reverseOrder(): Comparator<T> = @Suppress("UNCHECKED_CAST") (ReverseOrderComparator as Comparator<T>)

private class ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {
    override fun compare(a: T, b: T): Int = comparator.compare(b, a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<T> = comparator
}

private object NaturalOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = a.compareTo(b)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = ReverseOrderComparator
}

private object ReverseOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = b.compareTo(a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator
}
package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the
// GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.js.*

/**
 * Returns the greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 * If either value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR", "NOTHING_TO_INLINE")
public actual inline fun maxOf(a: Byte, b: Byte): Byte {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the greater of two values.
 * If either value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Short>()
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int): Int {
    return Math.max(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Long, b: Long): Long {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 * If either value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Float, b: Float): Float {
    return Math.max(a, b)
}

/**
 * Returns the greater of two values.
 * If either value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Double, b: Double): Double {
    return if (a >= b) a else b
}

/**
 * Returns the greater of three values.
 * If there are multiple equal maximal values, returns the first of them.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, vararg other: T): T {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Byte, vararg other: Byte): Byte {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Short, vararg other: Short): Short {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Int, vararg other: Int): Int {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Long, vararg other: Long): Long {
    return maxOf(a, maxOf(vararg other))
}

/**
 * Returns the greater of three values.
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Float, vararg other: Float): Float {
    return Math.max(a, vararg other)
}

/**
 * Returns the greater of three values.
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual fun maxOf(a: Double, vararg other: Double): Double {
    return Math.max(a, vararg other)
}

/**
 * Returns the greater of the given values.
 * If there are multiple equal maximal values, returns the first of them.
 */
@SinceKotlin("1.4")
public actual fun <T : Comparable<T>> maxOf(a: T, vararg other: T): T {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Byte, vararg other: Byte): Byte {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Short, vararg other: Short): Short {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Int, vararg other: Int): Int {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}
/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Long, vararg other: Long): Long {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Float, vararg other: Float): Float {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public actual fun maxOf(a: Double, vararg other: Double): Double {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Byte, b: Byte): Byte {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Short, b: Short): Short {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Short>()
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Int, b: Int): Int {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Long, b: Long): Long {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: T, b: T, c: T): T {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Byte, b: Byte, c: Byte): Byte {
    return Math.min(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Short, b: Short, c: Short): Short {
    return Math.min(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Short>()
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Int, b: Int, c: Int): Int {
    return Math.min(a, b, c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Long, b: Long, c: Long): Long {
    return Math.min(a, Math.min(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Float, b: Float, c: Float): Float {
    return Math.min(a, Math.min(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public actual fun minOf(a: Double, b: Double, c: Double): Double {
    return Math.min(a, Math.min(b, c))
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
public actual fun <T : Comparable<T>> minOf(a: T, b: T): T {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
public actual fun <T : Comparable<T>> minOf(a: T, vararg other: T): T {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Byte, vararg other: Byte): Byte {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Short, vararg other: Short): Short {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Int, vararg other: Int): Int {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Long, vararg other: Long): Long {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*/

/**
 * Returns the smaller of the given values.
 *\@SinceKotlin("1.4")\npublic actual fun minOf(a: Float, vararg other: Float): Float {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Double, vararg other: Double): Double {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.4")
public actual fun minOf(a: Byte, vararg other: Byte): Byte {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Short, vararg other: Short): Short {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Int, vararg other: Int): Int {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Long, vararg other: Long): Long {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*/

/**
 * Returns the smaller of the given values.
 *\@SinceKotlin("1.4")\npublic actual fun minOf(a: Float, vararg other: Float): Float {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
* Returns the smaller of the given values.
*\@SinceKotlin("1.4")\npublic actual fun minOf(a: Double, vararg other: Double): Double {\n    var min = a\n    for (e in other) min = minOf(min, e)\n    return min\n}/**
this.minus(other.toULong())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = ULong(this.data.minus(other.data))

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = this.times(other.toULong())

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UInt): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong = this.div(other.toULong())

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.rem(other.toULong())

    /** Increments this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): ULong = ULong(data.inc())

    /** Decrements this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): ULong = ULong(data.dec())

    /** Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)

    /** Shifts this value left by the [bitCount] number of bits. */
    @kotlin.internal.InlineOnly
    public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)

    /** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */
    @kotlin.internal.InlineOnly
    public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)

    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)

    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)

    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)

    /** Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): ULong = ULong(data.inv())

    /** Converts this [Byte] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline fun toByte(): Byte = data.toByte()

    /** Converts this [Short] value to [Short]. */
    @kotlin.internal.InlineOnly
    public inline fun toShort(): Short = data.toShort()

    /** Converts this [Int] value to [Int]. */
    @kotlin.internal.InlineOnly
    public inline fun toInt(): Int = data.toInt()
Converts this [ULong] value to [Long].

* If this value is less than or equals to [Long.MAX_VALUE], the resulting `Long` value represents the same numerical value as this `ULong`. Otherwise the result is negative.

* The resulting `Long` value has the same binary representation as this `ULong` value.

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data

/**
 * Converts this [ULong] value to [UByte].
 *
 * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `ULong`.
 *
 * The resulting `UByte` value is represented by the least significant 8 bits of this `ULong` value.
 */
@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/**
 * Converts this [ULong] value to [UShort].
 *
 * If this value is less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents the same numerical value as this `ULong`.
 *
 * The resulting `UShort` value is represented by the least significant 16 bits of this `ULong` value.
 */
@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/**
 * Converts this [ULong] value to [UInt].
 *
 * If this value is less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `ULong`.
 *
 * The resulting `UInt` value is represented by the least significant 32 bits of this `ULong` value.
 */
@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Returns this value. */
@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = this

/**
 * Converts this [Byte] value to [ULong].
 *
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Byte` value.
 *
 * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `Byte` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Short] value to [ULong].
 *
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Short` value.
 *
 * The least significant 16 bits of the resulting `ULong` value are the same as the bits of this `Short` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Int] value to [ULong].
 *
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Int` value.
 *
 * The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `Int` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Long] value to [ULong].
 *
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Long` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toULong(): ULong = ULong(this)

/**
 * Converts this [Float] value to [ULong].
 *
 * The fractional part, if any, is rounded down towards zero.
 *
 * Returns zero if this `Float` value is negative or `NaN`, [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

/**
 * Converts this [Double] value to [ULong].
 *
 * The fractional part, if any, is rounded down towards zero.
 *
 * Returns zero if this `Double` value is negative or `NaN`, [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toULong(): ULong = doubleToULong(this.toDouble())
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns 1st *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 1.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component1(): T = get(0)

/**
 * Returns 2nd *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 2.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component2(): T = get(1)

/**
 * Returns 3rd *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 3.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component3(): T = get(2)

/**
 * Returns 4th *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 4.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component4(): T = get(3)

/**
 * Returns 5th *element* from the list.
 * 
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 5.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component5(): T = get(4)

/**
 * Returns an element at the given *index* or throws an [IndexOutOfBoundsException] if the *index* is out of bounds of this collection.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
public fun <T> Iterable<T>.elementAt(index: Int): T {
    if (this is List)
        return get(index)
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("Collection doesn't contain element at index $index.") }
}

/**
 * Returns an element at the given *index* or the result of calling the [defaultValue] function if the *index* is out of bounds of this collection.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
public fun <T> Iterable<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    if (this is List)
        return this.getOrElse(index, defaultValue)
    if (index < 0)
        return null
    val iterator = iterator()
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return defaultValue(index)
}

/**
 * Returns an element at the given *index* or `null` if the *index* is out of bounds of this collection.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
public fun <T> Iterable<T>.elementAtOrNull(index: Int): T? {
    if (this is List)
        return getOrNull(index)
    if (index < 0)
        return null
    val iterator = iterator() // throw IndexOutOfBoundsException("Collection doesn't contain element at index $index.")
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return null
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such element was found.

```
Iterable<T>.find(predicate: (T) -> Boolean): T? {

    return firstOrNull(predicate)
}
```

```
Iterable<T>.findLast(predicate: (T) -> Boolean): T? {

    return lastOrNull(predicate)
}
```

```
List<T>.findLast(predicate: (T) -> Boolean): T? {

    return lastOrNull(predicate)
}
```

```
Iterable<T>.first(): T {

    when (this) {
        is List -> return this.first()
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                throw NoSuchElementException("Collection is empty.")

            return iterator.next()
        }
    }
}
```

```
List<T>.first(): T {

    if (isEmpty())
    throw NoSuchElementException("List is empty.")

    return this[0]
}
```

```
Iterable<T>.firstOrNull(): T? {

    when (this) {
        is List -> {
            if (isEmpty())
                return null
            else
                return this[0]
        }
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null

            return iterator.next()
        }
    }
}
```

```
List<T>.firstOrNull(): T? {

    return if (isEmpty()) null else this[0]
}
```

```
Iterable<T>.firstOrNull(predicate: (T) -> Boolean): T? {

    for (element in this) if (predicate(element)) return element

    throw NoSuchElementException("Collection contains no element matching the predicate.")
}
```

```
Iterable<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {

    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```
List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {

    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```
Iterable<T>.indexOf(element: T): Int {

    if (this is List) return this.indexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}
```

```
List<T>.indexOf(element: T): Int {

    return indexOf(element)
}
```

```
Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {

    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

```
List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {

    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}
```

```
Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {

    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return lastIndex
}
```

```
List<T>.indexOfLast(predicate: (T) -> Boolean): Int {

    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return lastIndex
}
```
the given [predicate], or -1 if the list does not contain such element.

List<T>.indexOfLast(predicate: (T) -> Boolean): Int { 
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) { 
        if (predicate(iterator.previous())) 
            return iterator.nextIndex()
    } 
    return -1
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) {
        val element = iterator.previous()
        if (predicate(element)) return element
    }
    throw NoSuchElementException("List contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) {
        val element = iterator.previous()
        if (element == element) return iterator.nextIndex()
    }
    throw NoSuchElementException("List contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
@Suppress("UNCHECKED_CAST") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns the last element, or `null` if the collection is empty.
 */
public fun <T> Iterable<T>.lastOrNull(): T? {
    when (this) {
        is List -> return if (isEmpty()) null else this[size - 1]
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null
            var last = iterator.next()
            while (iterator.hasNext())
                last = iterator.next()
            return last
        }
    }
}

/**
 * Returns the last element, or `null` if the list is empty.
 */
public fun <T> List<T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.random(): T {
    return random(Random)
}

the given [predicate], or -1 if the list does not contain such element.

List<T>.indexOfLast(predicate: (T) -> Boolean): Int { 
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) { 
        if (predicate(iterator.previous())) 
            return iterator.nextIndex()
    } 
    return -1
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) {
        val element = iterator.previous()
        if (predicate(element)) return element
    }
    throw NoSuchElementException("List contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns the last element, or `null` if the collection does not contain element.
 */
public fun <T> Iterable<T>.lastOrNull(): T? {
    when (this) {
        is List -> return if (isEmpty()) null else this[size - 1]
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null
            var last = iterator.next()
            while (iterator.hasNext())
                last = iterator.next()
            return last
        }
    }
}

/**
 * Returns the last element, or `null` if the list is empty.
 */
public fun <T> List<T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}

the given [predicate], or -1 if the list does not contain such element.

List<T>.indexOfLast(predicate: (T) -> Boolean): Int { 
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) { 
        if (predicate(iterator.previous())) 
            return iterator.nextIndex()
    } 
    return -1
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) {
        val element = iterator.previous()
        if (predicate(element)) return element
    }
    throw NoSuchElementException("List contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns the last element, or `null` if the collection does not contain element.
 */
public fun <T> Iterable<T>.lastOrNull(): T? {
    when (this) {
        is List -> return if (isEmpty()) null else this[size - 1]
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null
            var last = iterator.next()
            while (iterator.hasNext())
                last = iterator.next()
            return last
        }
    }
}

/**
 * Returns the last element, or `null` if the list is empty.
 */
public fun <T> List<T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}

the given [predicate], or -1 if the list does not contain such element.

List<T>.indexOfLast(predicate: (T) -> Boolean): Int { 
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) { 
        if (predicate(iterator.previous())) 
            return iterator.nextIndex()
    } 
    return -1
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun <T> List<T>.last(predicate: (T) -> Boolean): T {
    val iterator = this.listIterator(size)
    while (iterator.hasPrevious()) {
        val element = iterator.previous()
        if (predicate(element)) return element
    }
    throw NoSuchElementException("List contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}

/**
 * Returns the last element, or `null` if the collection does not contain element.
 */
public fun <T> Iterable<T>.lastOrNull(): T? {
    when (this) {
        is List -> return if (isEmpty()) null else this[size - 1]
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null
            var last = iterator.next()
            while (iterator.hasNext())
                last = iterator.next()
            return last
        }
    }
}

/**
 * Returns the last element, or `null` if the list is empty.
 */
public fun <T> List<T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}
/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Iterable<T>.filter(predicate: (T) -> Boolean): List<T> {
    return filterTo(ArrayList<T>(), predicate)
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <reified R> Iterable<*>.filterIsInstance(): List<R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Iterable<T>.filterIndexed(predicate: (Int, T) -> Boolean): List<T> {
    return filterIndexedTo(ArrayList<T>(), predicate)
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <reified R> Iterable<*>.filterIsInstanceTo(destination: C): C {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return this.subList(indices.start, indices.endInclusive + 1).toList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun <T> List<T>.slice(indices: Iterable<Int>): List<T> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<T>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.take(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (this is Collection<T>) {
        if (n >= size) return toList()
        if (n == 1) return listOf(first())
    }
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return optimizeReadOnlyList(list)
}

/**
 * Returns a list containing last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.takeLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (this is Collection<T>) {
        if (n >= size) return toList()
        if (n == 1) return listOfLast()
    }
    var varCount = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++varCount == n)
            break
    }
    return optimizeReadOnlyList(list)
}

/**
 * Returns a list containing all elements not null.
 */
public fun <T : Any> Iterable<T?>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())
}

/**
 * Appends all elements that are not null to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterNotNullTo(destination: C): C {
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <reified R> Iterable<*>.filterIsInstanceNull(): List<? : R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Appends all elements that are not null to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterIsInstanceNullTo(destination: C): C {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Returns a list containing all elements at indices in the specified [indices] range.
 */
public fun <T> List<T>.indices(): List<Int> {
    return listOf(*indices)
}

/**
 * Returns a list containing all elements at specified [indices].
 */
public fun <T> List<T>.indicesAt(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return this.subList(indices.start, indices.endInclusive + 1)
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun <T> List<T>.indicesAt(indices: Iterable<Int>): List<Int> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<Int>(size)
    for (index in indices) {
        list.add(index)
    }
    return list
}

/**
 * Returns a list containing all elements at indices in the specified [indices] range.
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return this.subList(indices.start, indices.endInclusive + 1).toList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun <T> List<T>.slice(indices: Iterable<Int>): List<T> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<T>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.take(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (this is Collection<T>) {
        if (n >= size) return toList()
        if (n == 1) return listOf(first())
    }
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return optimizeReadOnlyList(list)
}

/**
 * Returns a list containing last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.takeLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (this is Collection<T>) {
        if (n >= size) return toList()
        if (n == 1) return listOfLast()
    }
    var varCount = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++varCount == n)
            break
    }
    return optimizeReadOnlyList(list)
}

/**
 * Returns a list containing all elements not null.
 */
public fun <T : Any> Iterable<T?>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())
}

/**
 * Appends all elements that are not null to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterNotNullTo(destination: C): C {
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <reified R> Iterable<*>.filterIsInstanceNull(): List<? : R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Appends all elements that are not null to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Iterable<T?>.filterIsInstanceNullTo(destination: C): C {
    for (element in this) if (element is R) destination.add(element)
    return destination
}
/*
 * Open Source Used In DNAC Platform Arica  7921
 */
public fun <T> List<T>.takeLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size
    if (n >= size) return toList()
    if (n == 1) return listOf(last())
    val list = ArrayList<T>(n)
    if (this is RandomAccess) {
        for (index in size - n until size)
            list.add(this[index])
    } else {
        for (item in listIterator(size - n))
            list.add(item)
    }
    return list
}

/**
 * Returns a list containing last elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun <T> List<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {
    if (isEmpty())
        return emptyList()
    val iterator = listIterator(size)
    while (iterator.hasPrevious()) {
        if (!predicate(iterator.previous())) {
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
            if (expectedSize == 0) return emptyList()
            return ArrayList<T>(expectedSize).apply {
                while (iterator.hasNext())
                    add(iterator.next())
            }
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun <T> Iterable<T>.takeWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Reverses elements in the list in-place.
 */
public expect fun <T> MutableList<T>.reverse(): Unit

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Iterable<T>.reversed(): List<T> {
    if (this is Collection && size <= 1) return toList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Randomly shuffles elements in this list in-place
 * using the specified [random] instance as the source of randomness.
 * 
 * @sample samples.collections.Collections.Transformations.shuffle
 */
@SinceKotlin("1.3")
public fun <T> MutableList<T>.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        this[j] = this.set(i, this[j])
    }
}

/**
 * Sorts elements in the list in-place according to natural sort order
 * of the value returned by specified [selector] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative
 * to each other after sorting.
 */
public inline fun <T, R : Comparable<R>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareBy(selector))
}

/**
 * Sorts elements in the list in-place descending according to natural sort order
 * of the value returned by specified [selector] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative
 * to each other after sorting.
 */
public inline fun <T, R : Comparable<R>> MutableList<T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareByDescending(selector))
}

/**
 * Sorts elements in the list in-place descending according to their natural sort order.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative
 * to each other after sorting.
 */
public fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {
    sortWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative
 * to each other after sorting.
 */
public fun <T : Comparable<T>> Iterable<T>.sortedDescending(): List<T> {
    return toMutableList().apply { sort() }
}

/**
 * Returns a list of all elements sorted according to natural sort order
 * of the value returned by specified [selector] function.
 * 
 * @sample samples.collections.Collections.Sorting.sortedBy
 */
public inline fun <T, R : Comparable<R>> Iterable<T>.sortedBy(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareBy(selector))
}

/**
 * Returns a list of all elements sorted descending according to natural sort order
 * of the value returned by specified [selector] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative
 * to each other after sorting.
 */
public fun <T : Comparable<T>> Iterable<T>.sortedDescending(): List<T> {
    return sortedWith(compareByDescending(selector))
}
Open Source Used In DNAC Platform Arica 7922

```kotlin
{
    return sortedWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to the specified
 * comparator. The sort is *stable*. It means that equal elements preserve their order relative to each other
 * after sorting. */
fun <T> List<T>.sortedWith(comparator: Comparator<T>): List<T> {
    if (this is Collection) {
        if (size <= 1) return this.toList()
        return sortedWith(comparator).asList()
    } else
        return toMutableList().apply { sortWith(comparator) }.asList()
}

/**
 * Returns an array of Boolean containing all of the
 * elements of this collection. */
fun BooleanCollection.toBooleanArray(): BooleanArray {
    val result = BooleanArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Byte containing all of the
 * elements of this collection. */
fun ByteCollection.toByteArray(): ByteArray {
    val result = ByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Char containing all of the
 * elements of this collection. */
fun CharCollection.toCharArray(): CharArray {
    val result = CharArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Double containing all of the
 * elements of this collection. */
fun DoubleCollection.toDoubleArray(): DoubleArray {
    val result = DoubleArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Float containing
 * all of the elements of this collection. */
fun FloatCollection.toFloatArray(): FloatArray {
    val result = FloatArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Int containing all of the
 * elements of this collection. */
fun IntCollection.toIntArray(): IntArray {
    val result = IntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Long containing all of the
 * elements of this collection. */
fun LongCollection.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Short containing all of the
 * elements of this collection. */
fun ShortCollection.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
/**
 * Returns a [Map] containing key-value pairs
 * provided by [transform] function
 * applied to elements of the given collection.
 * If any of two pairs would have the same key
 * the last one gets added to the map. The
 * returned map preserves the entry iteration order
 * of the original collection. */
inline fun <T, K, V> Iterable<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/**
 * Returns a [Map] containing the elements from the given collection indexed
 * by the key returned from [keySelector] function applied to each element. If any two elements would have the same key
 * the last one gets added to the map. The returned map preserves the entry iteration order
 * of the original collection. */
inline fun <T, K> Iterable<T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
}

/**
 * Returns a [Map] containing the values provided by [valueTransform] and
 * indexed by [keySelector] functions applied to elements of the given collection. If any two elements would have the same key returned by
 * [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order
 * of the original collection. */
inline fun <T, K, V> Iterable<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

```
Open Source Used In DNAC Platform Arica  7923

```kotlin
val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
```

If any two elements are equal, the last one overwrites the former value in the map.

```kotlin
val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
```

If any two elements are equal, the last one overwrites the former value in the map.

```kotlin
val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
```

If any two elements are equal, the last one overwrites the former value in the map.

```kotlin
val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
```

If any two elements are equal, the last one overwrites the former value in the map.
transform
)\n
\n
* Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.\n
* @sample samples.collections.Collections.Transformations.flatMap

\n
\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapSequence")\npublic inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Sequence<R>): List<R> {\n    return flatMapTo(ArrayList<R>(), transform)\n}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection.\n
* @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterable")\n@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {\n    return flatMapIndexedTo(ArrayList<R>(), transform)\n}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection, to the given \n * [destination].\n
* @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedSequence")\n@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): List<R> {\n    return flatMapIndexedTo(ArrayList<R>(), transform)\n}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection, to the given \n * [destination].\n
* @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(checkIndexOverflow(index++), element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element\n * and its index in the original collection, to the given \n * [destination].\n
* @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedSequenceTo")\n@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {\n    var index = 0\n    for (element in this) {\n        val list = transform(checkIndexOverflow(index++), element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**
 * Groups elements of the original collection by the key returned by the given \n * [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of \n * corresponding elements.\n
* @sample samples.collections.Collections.Transformations.groupBy\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("groupBy")\npublic inline fun <T, K> Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {\n    return
groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original collection.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        destination.getOrPut(key) { ArrayList<T>() }.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a collection to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @return Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <T, R> Iterable<T>.map(transform: (T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapIndexedNotNullTo(destination: C, keySelector: (T) -> K, valueTransform: (T) -> V): C {
    for (index in this) {
        var element = this[index]
        if (element != null) {
            var key = keySelector(element)
            var list = destination.getOrPut(key) { ArrayList<T>() }
            list.add(element)
        }
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a collection to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @return Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <T, R> Iterable<T>.map(transform: (T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapIndexedNotNullTo(destination: C, keySelector: (T) -> K, valueTransform: (T) -> V): C {
    for (index in this) {
        var element = this[index]
        if (element != null) {
            var key = keySelector(element)
            var list = destination.getOrPut(key) { ArrayList<T>() }
            list.add(element)
        }
    }
    return destination
}
MutableCollection<R> mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(checkIndexOverflow(index++), item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given function to each element in the original collection.
 * @sample samples.collections.Transformations.mapTransform
 */
public fun <T, R> Iterable<T>.mapNotNull(transform: (T) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

/**
 * Applies the given function to each element in the original collection and appends the results to the given destination.
 * @sample samples.collections.Transformations.map
 */
public fun <T, R, C : MutableCollection<in R>> Iterable<T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original collection into an [IndexedValue] containing the index of that element and the element itself.
 * @sample samples.collections.Transformations.withIndex
 */
public fun <T> Iterable<T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a list containing only distinct elements from the given collection.
 * @sample samples.collections.Collections.Transformations.distinct
 */
public fun <T> Iterable<T>.distinct(): List<T> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only elements from the given collection having distinct keys returned by the given selector function.
 * @sample samples.collections.Collections.Transformations.distinctBy
 */
public inline fun <T, K> Iterable<T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a set containing all elements that are contained by both this collection and the specified collection.
 * @sample samples.collections.Collections.Transformations.union
 */
public infix fun <T> Iterable<T>.union(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns `true` if all elements match the given predicate.
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun <T> Iterable<T>.all(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return true
    for (element in this)
        if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if collection has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun <T> Iterable<T>.any(): Boolean {
    if (this is Collection) return !isEmpty()
    return iterator().hasNext()
}
* Returns `true` if at least one element matches the given [predicate].

```kotlin
public inline fun <T> Iterable<T>.any(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}
```

* Returns the number of elements in this collection.

```kotlin
public fun <T> Iterable<T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) count++
    return count
}
```

* Returns the number of elements in this collection.

```kotlin
public fun <T> Collection<T>.count(): Int {
    return size
}
```

* Accumulates value starting with [initial] value and applying [operation] from left to right to each element.

```kotlin
public inline fun <T, R> Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from right to left to each element.

```kotlin
public inline fun <T, R> List<T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var accumulator = initial
    if (!isEmpty()) {
        val iterator = listIterator(size)
        while (iterator.hasPrevious()) {
            accumulator = operation(iterator.previous(), accumulator)
        }
    }
    return accumulator
}
```

* Performs the given [action] on each element.

```kotlin
public inline fun <T> Iterable<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}
```

* Performs the given [action] on each element with sequential index.

```kotlin
public inline fun <T> Iterable<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

* Performs the given [action] on each element with sequential index and element itself.

```kotlin
fun Iterable<Double>.max(): Double? {
    return maxOrNull()
}
```

* Performs the given [action] on each element with sequential index and element itself.

```kotlin
fun Iterable<Float>.max(): Float? {
    return maxOrNull()
}
```

* Performs the given [action] on each element with sequential index and element itself.

```kotlin
fun <T : Comparable<T>> Iterable<T>.max(): T? {
    return maxByOrNull()
}
```
instead.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Iterable<T>.maxByOrNull(selector: (T) -> R): T? {
    return maxByOrNull(selector) ?: return null
}
```

- Instead.
- ReplaceWith("maxByOrNull(selector)")
- @DeprecatedSinceKotlin(warningSince = "$1.4")
- @OptIn(kotlin.experimental.ExperimentalTypeInference::class)
- @OverloadResolution
- @kotlin.internal.InlineOnly
- public inline fun <T, R : Comparable<R>> Iterable<T>.maxByOrNull(selector: (T) -> R): T? {
    return maxByOrNull(selector) ?: return null
}

- Returns the first element yielding the largest value of the given function or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public inline fun <T, R : Comparable<R>> Iterable<T>.maxByOrNull(selector: (T) -> R): T? {
    val iterator = iterator() ?: return null
    var maxElem = iterator.next() ?: return null
    var maxVal = selector(maxElem) ?: return null
    var maxValue = maxVal
    do {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxElem = v
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection. * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. * @throws NoSuchElementException if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOf(selector: (T) -> Double): Double {
    val iterator = iterator() ?: throw NoSuchElementException()
    var maxValue = selector(iterator.next()) ?: return null
    while (iterator.hasNext()) {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection. * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. * @throws NoSuchElementException if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOf(selector: (T) -> Float): Float {
    val iterator = iterator() ?: throw NoSuchElementException()
    var maxValue = selector(iterator.next()) ?: return null
    while (iterator.hasNext()) {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection. * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. * @throws NoSuchElementException if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOf(selector: (T) -> Float): Float {
    val iterator = iterator() ?: throw NoSuchElementException()
    var maxValue = selector(iterator.next()) ?: return null
    while (iterator.hasNext()) {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements. * @throws NoSuchElementException if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator() ?: return null
    var maxVal = selector(iterator.next()) ?: return null
    var maxValue = maxVal
    do {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxVal = v
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxVal
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements. * @throws NoSuchElementException if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator() ?: return null
    var maxVal = selector(iterator.next()) ?: return null
    var maxValue = maxVal
    do {
        val v = selector(iterator.next()) ?: return null
        if (maxValue < v) {
            maxVal = v
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxVal
}
```

- Returns the largest value among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements.
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@n\@public inline fun <T, R : Comparable<R>>
Iterable<T>.maxOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>>
Iterable<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [selector] function applied to each element in the collection or null if there are no elements.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@n\@public inline fun <T, R : Comparable<R>>
Iterable<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest element or null if there are no elements.
 */
@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <T : Comparable<T>>
Iterable<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.", ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Iterable<T>.maxWith(comparator: Comparator<in T>): T? {
    return maxWithOrNull(comparator)
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or null if there are no elements.
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.min(): Double? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Iterable<Float>.min(): Float? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Iterable<T>.minByOrNull(): T? {
    return minByOrNull()
}
inline fun <T, R : Comparable<R>> Iterable<T>.minBy(selector: (T) -> R): T? {
    return
    minByOrNull(selector)
}

/**
 * Returns the first element yielding the smallest value of the given function or
 * `null` if there are no elements.
 */
@sample samples.collections.Collections.Aggregates.minByOrNull
@SinceKotlin("1.4")
public in
line fun <T, R : Comparable<R>> Iterable<T>.minByOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    var minValue = selector(minElem)
    var e = iterator.next()
    var v = selector(e)
    while (iterator.hasNext()) {
        if (minValue > v) {
            minElem = e
            minValue = v
        }
        val elem = iterator.next()
        val elemValue = selector(elem)
        if (minValue > elemValue) {
            minElem = elem
            minValue = elemValue
        }
    }
    return minElem
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T, R : Comparable<R>>
Iterable<T>.minOf(selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>>
Iterable<T>.minOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the collection or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun Iterable<Double>.minOrNull(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun Iterable<Float>.minOrNull(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>>
Iterable<T>.minOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Iterable<T>.minWith(comparator: Comparator<in T>): T? {
    return minWithOrNull(comparator)
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun <T> Iterable<T>.minWithOrNull(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (this is Collection) return isEmpty()
    return !iterator().hasNext()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Iterable<T>.none(predicate: (T) -> Boolean): Boolean {
    if (this is Collection) return isEmpty()
    return !iterator().hasNext()
        \n        * @param [action] function that takes the index of an element and the
element itself and performs the action on the element.\n
```kotlin
@SinceKotlin("1.4")
open inline fun <T, C : Iterable<T>> C.onEachIndexed(action: (index: Int, T) -> Unit): C {
    return apply {
        for_each_indexed(action)
    }
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n *\n * @param [operation] function that takes current accumulator value and an element,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun <T, C : Iterable<T>> C.reduce(operation: (acc: T, C) -> S): S {
    var accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)\n    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original collection.\n *\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduceIndexed\n */
public inline fun <T, C : Iterable<T>> C.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    var index = 1, accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(check_index_overflow(index++), accumulator, iterator.next())\n    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element.\n *\n * @param [operation] function that takes current accumulator value and an element,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, C : Iterable<T>> C.reduceOrNull(operation: (acc: T, C) -> S): S? {
    var accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)\n    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left\n * to current accumulator value and each element.\n *\n * @param [operation] function that takes current accumulator value and an element,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduceRight\n */
public inline fun <T, C : Iterable<T>> C.reduceRight(operation: (acc: T, C) -> S): S {
    var iterator = listIterator(size)\n    while (!iterator.hasPrevious()) \n        throw UnsupportedOperationException("Empty list can't be reduced.")\n    var accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)\n    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left\n * to current accumulator value and each element with its index in the original list.\n *\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduceRightIndexed\n */
public inline fun <T, C : Iterable<T>> List<T>.reduceRightIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()\n    while (!iterator.hasNext()) \n        throw UnsupportedOperationException("Empty list can't be reduced.")\n    var accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(check_index_overflow(index++), iterator.next(), accumulator)\n    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left\n * to current accumulator value and each element.\n *\n * @param [operation] function that takes current accumulator value and an element,\n *\n * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, C :Iterable<T>> List<T>.reduceRightOrNull(operation: (acc: T, C) -> S): S? {
    val iterator = this.iterator()\n    while (!iterator.hasNext()) \n        throw UnsupportedOperationException("Empty list can't be reduced.")\n    var accumulator = operation.accumulator, iterator = this.iterator()\n    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)\n    }
    return accumulator
}
```
* @param [operation] function that takes the index of an element, the element itself and current accumulator
value,\n* and calculates the next accumulator value.\n* @sample
samples.collections.Collections.Aggregates.reduceRight\n*\n@public inline fun <S, T : S>
List<T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {\n    var accumulator = operation(index, \n    \n    if (\n        iterator.hasNext()\n    ) \n        return accumulator;\n
    accumulator = \n    } \n    return accumulator;\n* Accumulates value starting with the last element and applying [operation] from right to \n    \n    @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* and calculates the next accumulator value.\n* @sample
samples.collections.Collections.Aggregates.reduceRightOrNull\n*\n@public inline fun <S, T : S> List<T>.reduceRightIndexedOrNull(operation: (index: Int, T, acc: S) -> S): S? {\n    \n    \n    \n    \n    \n    \n    accumulator = \n    \n    \n    accumulator = \n    \n    \n    accumulator = \n    \n    \n    } \n    return accumulator;\n* Returns `null` if the list is empty.\n
* @param [operation] function that takes the index of an element and current accumulator value,\n* and calculates the next accumulator value.\n* @sample
samples.collections.Collections.Aggregates.reduceRightOrNull\n*\n@WasExperimental(ExperimentalStdlibApi::class)\n@public inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {\n    \n    \n    \n    \n    \n    \n    \n    } \n    return accumulator;\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* to each element and current accumulator value that starts with [initial] value.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @sample
samples.collections.Collections.Aggregates.runningFold\n*\n@public inline fun <T, R> \nIterable<T>.runningFold(initial: R, operation: (acc: R, T) -> R): List<R> {\n    \n    \n    \n    \n    \n    } \n    return result;\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* to each element, its index in the original collection and current accumulator value that starts with \n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @param [operation] function that takes the index of an element, current accumulator value and an\n* element, and calculates the next accumulator value.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @sample
samples.collections.Collections.Aggregates.runningFoldIndexed\n*\n@public inline fun <T, R> \nIterable<T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {\n    \n    \n    \n    \n    \n    } \n    return result;\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* to each element and current accumulator value that starts with the first element of this collection.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @param [operation] function that takes the index of an element and current accumulator value,\n* and calculates the next accumulator value.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @sample
samples.collections.Collections.Aggregates.runningFoldIndexed\n*\n@public inline fun <T, R> \nIterable<T>.runningFoldIndexed(initial: R, operation: (Int, acc: R, T) -> R): List<R> {\n    \n    \n    \n    \n    \n    } \n    return result;\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* to each element and current accumulator value that starts with the first element of this collection.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @param [operation] function that takes the index of an element and current accumulator value,\n* and calculates the next accumulator value.\n* \n* Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n* \n* @sample
value and the element, and calculates the next accumulator value.
@sample

```kotlin
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply {
        add(accumulator)
    }
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.4")
public inline fun <S, T : S> Iterable<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply {
        add(accumulator)
    }
    var index = 1
    while (iterator.hasNext()) {
        accumulator = operation(index++, accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    return runningFold(initial, operation)
}
```

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

```kotlin
@Deprecated("Use runningReduce instead.", ReplaceWith("runningReduce(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
    return runningReduce(operation)
}
```

```kotlin
@Deprecated("Use runningReduceIndexed instead.", ReplaceWith("runningReduceIndexed(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    return runningReduceIndexed(operation)
}
```

```kotlin
public inline fun <T> Iterable<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

```kotlin
public inline fun <T> Iterable<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.
 */
public fun <T : Any> Iterable<T?>.requireNoNulls(): Iterable<T> {
    for (element in this) {
        if (element == null) {
            throw IllegalArgumentException("null element found in $this.")
        }
    }
    @Suppress("UNCHECKED_CAST")
    return this as Iterable<T>
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.
 */
public fun <T : Any> List<T?>.requireNoNulls(): List<T> {
    for (element in this) {
        if (element == null) {
            throw IllegalArgumentException("null element found in $this.")
        }
    }
    @Suppress("UNCHECKED_CAST")
    return this as List<T>
}

/**
 * Splits this collection into a list of lists each not exceeding the given [size].
 * The last list in the resulting list may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 */
public fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this collection into several lists each not exceeding the given [size] and applies the given [transform]
 * function to an each list.
 * The last list in the resulting list may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 */
public fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a list containing all elements of the original collection without the first occurrence of the given [element].
 */
public operator fun <T> Iterable<T>.minus(element: T): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    var removed = false
    return this.filterTo(result) { if (!removed
```
collection and then all elements of the given [elements] sequence.

```kotlin
Collection<T>.plus(elements: Sequence<T>): List<T> { val result = ArrayList<T>(this.size + 10) result.addAll(this) result.addAll(elements) return result }/*
 */
public operator fun <T> Collection<T>.plus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>(this.size + 10)
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a list containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    val result = ArrayList<T>(this.size + 10)
    result.addAll(this)
    result.addAll(element)
    return result
}

/**
 * Returns a list containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    val result = ArrayList<T>(this.size + 10)
    result.addAll(this)
    result.addAll(element)
    return result
}

/**
 * Returns a list of snapshots of the window of the given [size] sliding along this collection with the given [step], where each snapshot is a list.
 * Several last lists may have less elements than the given [size].
 * @param size the window size
 * @param step the step size
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved
 * @param samples.collections.Sequences.Transformations.takeWindows
 */
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (windowSize < size && !partialWindows) break
            result.add(List(windowSize) { this[it + index] })
            index += step
        }
        return result
    }
    val result = ArrayList<List<T>>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
        result.add(it)
    }
    return result
}

/**
 * Returns a list of results of applying the given [transform] function to an each list representing a view over the window of the given [size]
 * sliding along this collection with the given [step].
 * Several last lists may have less elements than the given [size].
 * @param size the size of the window
 * @param step the step size
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved
 * @param samples.collections.Sequences.Operations.zipIterable
 */
public infix fun <T, R> Iterable<T>.zip(other: Array<R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of results of applying the given [transform] function to an each pair of elements.
 * Several last lists may have less elements than the given [size].
 * @param size the size of the window
 * @param step the step size
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved
 * @param samples.collections.Operations.zipIterableWithTransform
 */
public inline fun <T, R, V> Iterable<T>.zip(other: Array<R>, transform: (a: T, b: R) -> V): List<V> {
    return zip(other, transform) { a, b -> minOf(collectionSizeOrDefault(10), arraySize) } { var i = 0 for (element in this) { if (i == 0) {
```
Iterable<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns an immutable list containing only the specified object [element].
 */
@kotlin.jvm.JvmName("copyToByteArray")
public fun Collection<T>.copyToByteArray(): ByteArray {
    return if (collection.isEmpty()) emptyArray<Byte>() else
        copyToByteArray(collection)
}

/**
 * Returns an immutable list containing only the specified object [element].
 */
@kotlin.jvm.JvmName("listOf")
public fun <T> listOf(element: T): List<T> =
arrayListOf(element)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <E> buildListInternal(builderAction: MutableList<E>.() -> Unit): List<E> {
    return ArrayList<E>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <E> buildListInternal(capacity: Int, builderAction: MutableList<E>.() -> Unit): List<E> {
    return ArrayList<E>(capacity).apply(builderAction).build()
}

/**
 * Returns an immutable set containing only the specified object [element].
 */
public fun <T> setOf(element: T): Set<T> = hashSetOf(element)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <E> buildSetInternal(builderAction: MutableSet<E>.() -> Unit): Set<E> {
    return LinkedHashSet<E>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E> {
    return LinkedHashSet<E>(capacity).apply(builderAction).build()
}

/**
 * Returns an immutable map, mapping only the specified key to the specified value.
 */
public fun <K, V> mapOf(pair: Pair<K, V>): Map<K, V> = hashMapOf(pair)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <K, V> buildMapInternal(builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    return LinkedHashMap<K, V>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal:inlineOnly
internal actual inline fun <K, V> buildMapInternal(capacity: Int, builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    return LinkedHashMap<K, V>(capacity).apply(builderAction).build()
}

/**
 * Fills the list with the provided [value]. Each element in the list gets replaced with the [value].
 */
@SinceKotlin("1.2")
public actual fun <T> MutableList<T>.fill(value: T): Unit {
    for (index in 0..lastIndex) {
        this[index] = value
    }
}

/**
 * Randomly shuffles elements in this list.
 */
@SinceKotlin("1.2")
public actual fun <T> MutableList<T>.shuffle(): Unit = shuffle(Random)

/**
 * Returns a new list with the elements of this list randomly shuffled.
 */
@SinceKotlin("1.2")
public actual fun <T> Iterable<T>.shuffled(): List<T> = toMutableList().apply { shuffle() }

/**
 * Sorts elements in the list in-place according to their natural sort order.
 *
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public actual fun <T : Comparable<T>> MutableList<T>.sort(): Unit {
    sortWith(compareTo { a, b -> a.compareTo(b) })
}

/**
 * Sorts elements in the list in-place according to the order specified with [comparator].
 *
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public actual fun <T> MutableList<T>.sortWith(comparator: Comparator<in T>): Unit {
    sortWith { a, b -> comparator.compare(a, b) }
}

/**
 * Copies the elements from this array to the specified destination.
 */
@PublishedApi
@since("1.2")
internal actual fun <T> arrayOfNulls(reference: Array<T>, size: Int): Array<T> {
    return arrayOfNulls<Any>(size).unsafeCast<Array<T>>()
}

@PublishedApi
@since("1.3")
@JsName("arrayCopy")
internal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int) {
    val sourceSize = source.size
    val arrayCopy = destinationOffset + sourceSize
    val subrange = source.asDynamic().subarray(destinationOffset, arrayCopy)
    destination.asDynamic().set(subrange, destinationOffset)
rangeSize - 1 downTo 0) { 
    destination[destinationOffset + index] = source[startIndex + index]
}

@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <K, V> Map<K, V>.toSingletonMapOrSelf(): Map<K, V> = 
    this@toSingletonMapOrSelf

@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <K, V> Map<K, V>.toSingletonMap(): Map<K, V> = 
    this@toSingletonMapOrSelf

@Suppress("NOTHING_TO_INLINE")
internal actual inline fun <K, V> Map<K, V>.toMutableMap(): Map<K, V> = 
    this@toSingletonMapOrSelf

@PublishedApi
internal actual inline fun checkIndexOverflow(index: Int): Int {
    if (index < 0) {
        throw IndexOutOfBoundsException()
    }
    return index
}

@PublishedApi
internal actual inline fun checkCountOverflow(count: Int): Int {
    if (count < 0) {
        throw IndexOutOfBoundsException()
    }
    return count
}

/**
 * Returns the greatest of two values.
 * 
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public expect fun <T : Comparable<T>> maxOf(a: T, b: T): T

/**
 * Returns the greatest of three values.
 * 
 * If there are multiple equal maximal values, returns the first of them.
 */
@SinceKotlin("1.1")
public expect fun <T : Comparable<T>> maxOf(a: Byte, b: Byte, c: Byte): T

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

/**
 * Returns the given iterator builder function return value.
 */
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("ComparisonsKt")

package kotlin.comparisons


/*
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte

/*
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline fun maxOf(a: Short, b: Short, c: Short): Short

/*
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline fun maxOf(a: Int, b: Int, c: Int): Int

/*
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline fun maxOf(a: Long, b: Long, c: Long): Long

/*
 * Returns the greater of three values.
 * If any value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline
fun maxOf(a: Float, b: Float, c: Float): Float

/*
 * Returns the greater of three values.
 * If any value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public
expect inline
fun maxOf(a: Double, b: Double, c: Double): Double

/*
 * Returns the greater of three values according to the order
 * specified by the given [comparator].
 */
@SinceKotlin("1.4")
public
expect fun <T> maxOf(a: T, vararg other: T): T

/*
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public
expect fun maxOf(a: Byte, vararg other: Byte): Byte

/*
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public
expect fun maxOf(a: Short, vararg other: Short): Short

/*
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public
expect fun maxOf(a: Int, vararg other: Int): Int

/*
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public
expect fun maxOf(a: Long, vararg other: Long): Long

/*
 * Returns the greater of the given values.
 * If any value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.4")
public
expect inline
fun maxOf(a: Float, b: Float): Float

/*
 * Returns the greater of the given values.
 * If any value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.4")
public
expect inline
fun maxOf(a: Double, b: Double): Double

/*
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public
expect fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T

/*
 * Returns the smaller of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Byte, b: Byte): Byte

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Short, b: Short): Short

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Int, b: Int): Int

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Long, b: Long): Long

/*
 * Returns the smaller of two values.
 * If either value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Float, b: Float): Float

/*
 * Returns the smaller of two values.
 * If either value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
public
expect inline
fun minOf(a: Double, b: Double): Double

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Byte, b: Byte): Byte

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Short, b: Short): Short

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Int, b: Int): Int

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Long, b: Long): Long

/*
 * Returns the smaller of two values.
 * If either value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Float, b: Float): Float

/*
 * Returns the smaller of two values.
 * If either value is ‘NaN’, returns ‘NaN’.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Double, b: Double): Double

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Byte, b: Byte, c: Byte): Byte

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public
expect inline fun minOf(a: Short, b: Short, c: Short): Short
/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Int, b: Int, c: Int): Int

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Long, b: Long, c: Long): Long

/**
 * Returns the smaller of three values.
 * 
 * If any value is 
 * `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Float, b: Float, c: Float): Float

/**
 * Returns the smaller of three values.
 * 
 * If any value is 
 * `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Double, b: Double, c: Double): Double

/**
 * Returns the smaller of three values according to the order
 * specified by the given [comparator].
 */
public fun <T> minOf(a: T, b: T, c: T, comparator: Comparator<in T>): T {
    return minOf(a, minOf(b, c, comparator), comparator)
}

/**
 * Returns the smaller of two values according to the
 * order specified by the given [comparator].
 */
public fun <T> minOf(a: T, b: T, comparator: Comparator<in T>): T {
    return if (comparator.compare(a, b) <= 0) a else b
}

/**
 * Returns the smaller of the given values.
 * 
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
public expect fun <T> minOf(a: T, vararg other: T): T

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked
 * on each entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <K, V, R> Map<K, V>.flatMap(transform: (Map.Entry<K, V>) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapTo")
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a list of all elements yielded from results of [transform] function being invoked on each
 * entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMap")
public inline fun <K, V, R> Map<K, V>.flatMap(transform: (Map.Entry<K, V>) -> R): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a list of all elements yielded from results of [transform] function being invoked on each
 * entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapTo")
    return flatMapTo(result, transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked
 * on each entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <K, V, R> Map<K, V>.flatMapSequence(transform: (Map.Entry<K, V>) -> Sequence<R>): List<R> {
    return flatMapSequence(transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked
 * on each entry of original map.
 * 
 * @sample samples.collections.Collections.Transformations.flatMap
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMap")
public inline fun <K, V, R> Map<K, V>.flatMap(transform: (Map.Entry<K, V>) -> R): List<R> {
    return flatMap(transform)
}
entry of original map, to the given [destination].

/**
 * Returns a list containing the results of applying the given [transform]
 * function\n * to each entry in the original map.\n * @sample samples.collections.Maps.Transformations.mapTo
 */
public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.mapTo(destination: C, transform: (Map.Entry<K, V>) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns `true` if all entries match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun <K, V> Map<out K, V>.all(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if map has at least one entry.\n * @sample samples.collections.Collections.Aggregates.any\n */
public fun <K, V> Map<out K, V>.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns the number of entries matching the given [predicate].\n * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <K, V> Map<out K, V>.any(predicate: (Map.Entry<K, V>) -> Boolean): Int {
    if (isEmpty()) return 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of entries in this map.\n */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.count(): Int {
    return size
}

/**
 * Returns the number of entries matching the given [predicate].\n */
public inline fun <K, V> Map<out K, V>.count(predicate: (Map.Entry<K, V>) -> Boolean): Int {
    if (isEmpty()) return 0
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Performs the given [action] on each entry.\n */
@kotlin.internal.HidesMembers
public inline fun <K, V> Map<out K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {
    for (element in this) action(element)
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@kotlin.internal.InlineOnly
    return entries.maxByOrNull(selector)\n    .let { destination.add(it) }\n        destination.addAll(list)\n    return destination\n}

*/
public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Returns `true` if at least one entry matches the given [predicate].\n * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <K, V> Map<out K, V>.anyWithPredicate(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the largest value among all values produced by [selector]
 * function\n * applied to each entry in the map.\n * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
    return maxByOrNull(selector)\n    .let { destination.add(it) }\n        destination.addAll(list)\n    return destination\n}

*/
public inline fun <K, V, R, C : MutableCollection<in R>> Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
Returned result is 'NaN'.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <K, V> Map\<out K, V>\(\)\,.maxOf(selector: (Map.Entry\(\langle K, V\rangle \) \:- Double): Double \{|\n  return entries.maxOf(selector)\} |n\}/**\n* Returns the largest value among all values produced by [selector] function\n* applied to each entry in the map.\n* \* If any of values produced by [selector] function is 'NaN', the returned result is 'NaN'.\n*
*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly\@public inline fun <K, V> Map\<out K, V>\(\)\,.maxOf(selector: (Map.Entry\(\langle K, V\rangle \) \:- Float): Float \{|\n  return entries.maxOf(selector)\} |n\}/**\n* Returns the largest value among all values produced by [selector] function\n* applied to each entry in the map.\n* \* If any of values produced by [selector] function is 'NaN', the returned result is 'NaN'.\n*
V>.maxWithOrNull(comparator: Comparator<in Map.Entry<K, V>>)?: Map.Entry<K, V>? {
    return entries.maxWithOrNull(comparator)
}

ReplaceWith("minByOrNull(selector)")@DeprecatedSinceKotlin(warningSince = "1.4")
    return minByOrNull(selector)
}

/**
 * Returns the first entry yielding the smallest value of the given function
 * or `null` if there are no entries.
 * @throws NoSuchElementException if the map is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> Double): Double {
    return minOfOrNull(selector)
}

/**
 * Returns the smallest value among all values produced by [selector]
 * function applied to each entry in the map.
 * @throws NoSuchElementException if the map is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V, R : Comparable<R>> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> R): R? {
    return minOfOrNull(selector)
}

/**
 * Returns the smallest value according to the provided [comparator]
 * function applied to each entry in the map or `null` if there are no entries.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
    return minOfWith(comparator, selector)
}
are no entries.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution

ByLambdaReturnType\@inline public inline fun <K, V, R> Map\<out K, V>\.minOfWithOrNull\(comparator: Comparator\<in R>, selector: (Map.Entry\<K, V>\) -> R\): R?\{\n    return entries.minOfWithOrNull\(comparator, selector\)\n\}

\@Deprecated\("Use minWithOrNull instead.""); ReplaceWith\("minWithOrNull\(comparator\)\")\@DeprecatedSinceKotlin(warningSince = "1.4")\@inline public fun <K, V> Map\<out K, V>\.minWithOrNull\(comparator: Comparator\<in Map.Entry\<K, V>>\): Map.Entry\<K, V>?\{\n    return minWithOrNull\(comparator\)\n\}

/**
 * Returns the first entry having the smallest value according to the provided [comparator] or `null` if there are no entries.

*\@SinceKotlin("1.4")\@inline public inline fun <K, V> Map\<out K, V>\.minWith\(comparator: Comparator\<in Map.Entry\<K, V>>\): Map.Entry\<K, V>?\{\n    return minWith\(comparator\)\n\}

/**
 * Returns `true` if no entries match the given [predicate].
 *
 * @sample
* samples.collections.Collections.Aggregates.none
 */

public fun <K, V> Map\<out K, V>\.none(): Boolean\{\n    return isEmpty()\n\}

/**
 * Returns `true` if no entries match the given [predicate].
 *
 * @sample
* samples.collections.Collections.Aggregates.noneWithPredicate
 */

public inline fun <K, V> Map\<out K, V>\.none\(predicate: (Map.Entry\<K, V>) -> Boolean\): Boolean\{\n    if (isEmpty()) return true
    for (element in this) if (predicate(element)) return false
    return true
\}

/**
 * Performs the given [action] on each entry and returns the map itself afterwards.
 *
 * @sample
* samples.collections.Collections.Aggregates.onEach
 */

@SinceKotlin("1.1")
public inline fun <K, V, M : Map\<out K, V>> M.onEach\(action: (Map.Entry\<K, V>) -> Unit\): M\{\n    return apply { for (element in this) action(element) }\n\}

/**
 * Performs the given [action] on each entry, providing sequential index with the entry, and returns
 * the map itself afterwards.
 *
 * @param [action] function that takes the index of an entry and the entry itself and performs
 * the action on the entry.

*\@SinceKotlin("1.4")\@inline public inline fun <K, V> Map\<out K, V>\.onEachIndexed\(action: (index: Int, Map.Entry\<K, V>) -> Unit\): Unit\{\n    return apply { forEachIndexed\(action\) }\n\}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 *
 * @sample
* samples.collections.Collections.AsIterable
 */

public inline fun <K, V> Map\<out K, V>\.asIterable(): Iterable\<Map.Entry\<K, V>>\{\n    return entries\n\}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries when being iterated.
 *
 * @sample
* samples.collections.Collections.AsSequence
 */

public fun <K, V> Map\<out K, V>\.asSequence(): Sequence\<Map.Entry\<K, V>>\{\n    return entries.asSequence()\n\}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

package
kotlin.sequences

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*

/**
 * Returns `true` if [element] is found in the sequence.
 *
 * The operation is _terminal_.
 */

public operator fun <T> Sequence\<T>.contains\(element: T\): Boolean\{\n    return indexOf\(element\) >= 0
\}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the
 * [index] is out of bounds of this sequence.
 *
 * @sample
* samples.collections.Collections.Elements.elementAt
 */

public fun <T> Sequence\<T>.elementAt\(index: Int\): T\{\n    return elementAtOrElse\(index\) { throw IndexOutOfBoundsException\("Sequence doesn't contain element at index $index.\") }\n\}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
 * [index] is out of bounds of this sequence.
 *
 * The operation is _terminal_.
 */

public fun <T> Sequence\<T>.elementAtOrElse\(index: Int, defaultValue: (Int) -> T\): T\{\n    return elementAtOrElse\(index\) { defaultValue\(index\) }\n\}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 *
 * @sample
* samples.collections.Collections.AsIterable
 */

@kotlin.internal.InlineOnly
public inline fun <T> Iterable\<T>.forEach\(action: (T) -> Unit\): Unit\{\n    returnforEach\(action\)\n\}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries when being iterated.
 *
 * @sample
* samples.collections.Collections.AsSequence
 */

public fun <T> Sequence\<T>.asSequence(): Sequence\<T>\{\n    returnforEach\(action\)\n\}
Sequence<T>.elementAtOrNull(index: Int): T? {
    if (index < 0)
        return null
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * The operation is _terminal_.
 */
@sample
val iterator = iterator()

Sequence<T>.find(predicate: (T) -> Boolean): T? {
    return firstOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * The operation is _terminal_.
 */
val iterator = iterator()

Sequence<T>.indexOfLast(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * The operation is _terminal_.
 */
@throws
val iterator = iterator()

Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * The operation is _terminal_.
 */
@sample
val iterator = iterator()

Sequence<T>.findLast(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 * The operation is _terminal_.
 */
@sample
val iterator = iterator()

Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * The operation is _terminal_.
 */
@throws
val iterator = iterator()

Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}
samples.collections.Collections.Filtering.filterIsInstanceOf

```kotlin
public inline fun <reified R>
    Sequence<*>.filterIsInstance(): Sequence<@kotlin.internal.NoInfer R> {
@Suppress("UNCHECKED_CAST")
    return filter { it is R } as Sequence<R>
}

/**
 * Appends all elements that are instances of specified type parameter R to the given [destination].
 * The operation is _terminal_.

 * @sample samples.collections.Collections.Filtering.filterIsInstanceTo
 */
public inline fun
    <reified R, C : MutableCollection<in R>> Sequence<*>.filterIsInstanceTo(destination: C): C {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing all elements not matching the given [predicate].
 * The operation is _intermediate_ and _stateless_.

 * @sample samples.collections.Collections.Filtering.filter
 */
public fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, false, predicate)
}

/**
 * Returns a sequence containing all elements that are not `null`.
 * The operation is _intermediate_ and _stateless_.

 * @sample samples.collections.Collections.Filtering.filterNotNull
 */
public fun <T : Any> Sequence<T?>.filterNotNull(): Sequence<T> {
    @Suppress("UNCHECKED_CAST")
    return filterNot { it == null } as Sequence<T>
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * The operation is _terminal_.

 * @sample samples.collections.Collections.Filtering.filterTo
 */
public inline fun <T, C : MutableCollection<in T>>
    Sequence<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * The operation is _terminal_.

 * @sample samples.collections.Collections.Filtering.filterTo
 */
public inline fun <T, C : MutableCollection<in T>>
    Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing first [n] elements.
 * The operation is _intermediate_ and _stateless_.

 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Sequence<T>.take(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return when {
        n == 0 -> emptySequence()
        this is DropTakeSequence -> this.take(n)
        else -> TakeSequence(this, n)
    }
}

/**
 * Returns a sequence containing first elements satisfying the given [predicate].
 * The operation is _intermediate_ and _stateless_.

 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {
    return TakeWhileSequence(this, predicate)
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

 * @sample samples.collections.Collections.Transformations.sorted
 */
public fun <T : Comparable<T>> Sequence<T>.sorted(): Sequence<T> {
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this@sorted.toMutableList()
            sortedList.sort()
            return sortedList.iterator()
        }
    }
}

/**
 * Returns a sequence that yields elements of this sequence sorted descending according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

 * @sample samples.collections.Collections.Transformations.sortedByDescending
 */
public fun <T, R : Comparable<R>> Sequence<T>.sortedBy(selector: (T) -> R): Sequence<T> {
    return sortedWith(compareBy(selector))
}

/**
 * Returns a sequence that yields elements of this sequence sorted ascending according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

 * @sample samples.collections.Collections.Transformations.sortedBy
 */
public fun <T, R : Comparable<R>> Sequence<T>.sortedByAscending(selector: (T) -> R): Sequence<T> {
    return sortedWith(compareByAscending(selector))
}
```
sorted descending according to their natural sort order. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting. The operation is _intermediate_ and _stateful_.

```kotlin
public fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator]. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting. The operation is _intermediate_ and _stateful_.
 */
public fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T> {
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this@sortedWith.toMutableList()
            sortedList.sortWith(comparator)
            return sortedList.iterator()
        }
    }
}

/**
 * Returns a sequence containing key-value pairs provided by [transform] function applied to elements of the given sequence.
 * If any of two pairs would have the same key the last one gets added to the map. The returned map preserves the entry iteration order of the original sequence.
 * The operation is _terminal_.
 */
public inline fun <T, K, V, M : MutableMap<in K, V>> Sequence<T>.associateTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
```
the map. The returned map preserves the entry iteration order of the original sequence. The operation is _terminal_ and _stateless_.

```kotlin
public fun <T, C : MutableCollection<in T>> Sequence<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}
```

```kotlin
public fun <T> Sequence<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>())
}
```

```kotlin
public fun <T> Sequence<T>.toList(): List<T> {
    return this.toMutableList().optimizeReadOnlyList()
}
```

```kotlin
public fun <T> Sequence<T>.toMutableList(): MutableList<T> {
    return toCollection(ArrayList<T>())
}
```

```kotlin
public fun <T> Sequence<T>.toSet(): Set<T> {
    return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()
}
```

```kotlin
/**
 * Returns a single sequence of all elements yielded from results of [transform] function being invoked on each element and its index in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexed")
public fun <T, R>
    Sequence<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): Sequence<R> {
    return flatMapIndexed(this, transform, Sequence<R>::iterator)
}
```

```kotlin
/**
 * Returns a single sequence of all elements yielded from results of [transform] function being invoked on each element and its index in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexed")
public fun <T, R>
    Sequence<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): Sequence<R> {
    return flatMapIndexed(this, transform, Sequence<R>::iterator)
}
```

```kotlin
/**
 * Appends all elements yielded from results of [transform] function being invoked on each element and its index in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexed")
public fun <T, R>
    Sequence<T>.flatMapIndexed(transform: (index: Int, T) -> Sequence<R>): Sequence<R> {
    return flatMapIndexed(this, transform, Sequence<R>::iterator)
}
```
The operation is _terminal_.

*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterableTo")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element of the original sequence, to the given [destination].
 *
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedSequenceTo")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original sequence by the key returned by the given [keySelector]
 * function applied to each element and puts to the [destination] map each group key associated
 * with a list of corresponding elements.
 *
 * The returned map preserves the entry iteration order of the keys produced from the original
 * sequence.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("groupByTo")
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}
MutableMap<in K, MutableList<V>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to each element in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {
    return TransformingSequence(this, transform)
}

/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to each element and its index in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R): Sequence<R> {
    return TransformingIndexedSequence(this, transform)
}

/**
 * Returns a sequence containing only the non-null results of applying the given [transform] function
 * to each element and its index in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}

/**
 * Applies the given [transform] function to each element and its index in the original sequence
 * and appends only the non-null results to the given destination.
 * The operation is _intermediate_ and _stateless_.
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
    forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }
    return destination
}

/**
 * Applies the given [transform] function to each element in the original sequence
 * and appends only the non-null results to the given destination.
 * The operation is _intermediate_ and _stateless_.
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a sequence that wraps each element of the original sequence
 * into an [IndexedValue] containing the index of that element and the element itself.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.withIndex():
Sequence<IndexedValue<T>> {  
    return IndexingSequence(this)
}

/**
 * Returns a sequence containing only distinct elements from the given sequence.
 * The elements in the resulting sequence are in the same order as they were in the source sequence.
 * The operation is _intermediate_ and _stateful_.
 */
public fun <T> Sequence<T>.distinct(): Sequence<T> {  
    return this.distinctBy { it }
}

/**
 * Returns a sequence containing only elements from the given sequence having equal keys, only the first one will be present in the resulting sequence.
 * The elements in the resulting sequence are in the same order as they were in the source sequence.
 * The operation is _intermediate_ and _stateful_.
 */
public fun <T, K> Sequence<T>.distinctBy(selector: (T) -> K): Sequence<T> {  
    return DistinctSequence(this, selector)
}

/**
 * Returns a new [MutableSet] containing all distinct elements from the given sequence.
 * The returned set preserves the element iteration order of the original sequence.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {  
    val set = LinkedHashSet<T>()
    for (item in this) set.add(item)
    return set
}

/**
 * Returns `true` if all elements match the given [predicate].
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {  
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if sequence has at least one element.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.any(): Boolean {  
    return iterator().hasNext()
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.any(predicate: (T) -> Boolean): Boolean {  
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right.
 * Returns the specified [initial] value if the sequence is empty.
 * The operation is _terminal_.
 */
public inline fun <T, R> Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {  
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Performs the given [action] on each element.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.forEach(action: (T) -> Unit): Unit {  
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {  
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

@Deprecated("Use maxOrNull instead.",
Open Source Used In DNAC Platform Arica

ReplaceWith("maxOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun Sequence<Double>.max(): Double? { return maxOrNull() }@DeprecatedSinceKotlin(warningSince = "1.4")@public fun Sequence<Float>.max(): Float? { return maxOrNull() }@SinceKotlin("1.1")@fun <T : Comparable<T>> Sequence<T>.max(): T? { return maxByOrNull(selector) }@SinceKotlin("1.4")@public inline fun <T, R : Comparable<R>> Sequence<T>.maxBy(selector: (T) -> R): T? { return maxByOrNull(selector) }@SinceKotlin("1.4")@public inline fun <T> Sequence<T>.maxOf(selector: (T) -> Double): Double { return maxOf(maxValue, v) }@SinceKotlin("1.4")@public inline fun <T> Sequence<T>.maxOf(selector: (T) -> Float): Float { return maxOf(maxValue, v) }@SinceKotlin("1.4")@public inline fun <T, R : Comparable<R>> Sequence<T>.maxOf(selector: (T) -> R): R { return maxOf(maxValue, v) }@SinceKotlin("1.4")@public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Double): Double? { return maxOrNull() }@SinceKotlin("1.4")@public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Float): Float? { return maxOrNull() }
values produced by [selector] function is `NaN`, the returned result is `NaN`.

* The operation is _terminal_.

* @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution

**LambdaReturnType**

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the sequence
 * or `null` if there are no elements.
 *
 * The operation is _terminal_.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution

**LambdaReturnType**

@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>>

Sequence<T>.maxOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest element according to the provided
 * [comparator] among all values produced by [selector] function applied to each element in the sequence
 * or `null` if there are no elements.
 *
 * The operation is _terminal_.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution

**LambdaReturnType**

@kotlin.internal.InlineOnly
public inline fun <T : Comparable<T>>

Sequence<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.",
ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Sequence<T>.maxWith(comparator: Comparator<in T>): T? {
    return maxWithOrNull(comparator)
}

/**
 * Returns the first element having the largest value according to the
 * provided [comparator] or `null` if there are no elements.
 *
 * The operation is _terminal_.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution

**LambdaReturnType**

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxWithOrNull(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}
minOf(minValue, v)\n\nreturn minValue\n\n\n* Returns the smallest value among all values produced
by [selector] function\n* applied to each element in the sequence or 'null' if there are no elements.\n* \n* If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n* The operation is _terminal_.\n
*
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.experimental.InlineOnly
public fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Float?): Float? {\n\nval iterator = iterator()\nif (!iterator.hasNext()) return null\nval minValue = selector(iterator.next())\nwhile (iterator.hasNext()) {\nval v = selector(iterator.next())\nminValue = minOf(minValue, v)\n}\nreturn minValue\n\n\n* Returns the smallest value among all values produced
by [selector] function\n* applied to each element in the sequence or 'null' if there are no elements.\n* \n* The operation is _terminal_.\n
*
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.experimental.InlineOnly
public fun <T> Sequence<T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {\n\nval iterator = iterator()\nif (!iterator.hasNext()) throw NoSuchElementException()\nvar minValue = selector(iterator.next())\nwhile (iterator.hasNext()) {\nval v = iterator.next()\nif (comparator.compare(minValue, v) > 0) {\nminValue = v\n}\n}\nreturn minValue\n\n\n* Returns the smallest value among all values produced by [selector] function applied to each element in the sequence or 'null' if there are no elements.\n* \n* The operation is _terminal_.\n
*
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.experimental.InlineOnly
public fun <T> Sequence<T>.minWith(comparator: Comparator<in T>): T? {\n\nval iterator = iterator()\nif (!iterator.hasNext()) throw NoSuchElementException()\nvar min = iterator.next()\nwhile (iterator.hasNext()) {\nval e = iterator.next()\nif (comparator.compare(min, e) < 0) {\nmin = e\n}\n}\nreturn min\n\n\n* Returns the first element having the smallest value according to the provided [comparator] or 'null' if there are no elements.\n* \n* The operation is _terminal_.\n
*
@SinceKotlin("1.4")
public fun <T> Sequence<T>.minWithOrNull(comparator: Comparator<in T>): T? {
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        var e = iterator.next()
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the sequence has no elements.
 * The operation is _terminal_ and _stateless_.
 * @param [predicate] function that takes the index of an element and the element itself and
 * performs the action on the element.
 * @return `true` if no elements match the given [predicate], `false` otherwise.
 * The operation is _terminal_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun <T> Sequence<T>.none(): Boolean {
    return iterator().hasNext()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * The operation is _terminal_ and _stateless_.
 * @param [predicate] function that takes the index of an element, current accumulator value and the element itself
 * and calculates the next accumulator value.
 * @return `true` if no elements match the given [predicate], `false` otherwise.
 * The operation is _terminal_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.
 * The operation is _intermediate_ and _stateless_.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.onEach
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.onEach(action: (T) -> Unit): Sequence<T> {
    return mapIndexed { index, element ->
        action(index, element)
        element
    }
}

/**
 * Returns a sequence which performs the given [operation] from left to right on the current accumulator value and each element from left to right.
 * The operation is _terminal_.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * The operation is _terminal_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Sequence<T>.reduce(operation: (acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.")
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Returns a sequence which performs the given [operation] from left to right on the current accumulator value and each element from left to right when the index of an element and the element itself
 * and calculates the next accumulator value.
 * The operation is _terminal_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.reduceIndexed
 */
public inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++, accumulator, iterator.next()))
        index = index + 1
    }
    return accumulator
}

/**
 * Returns a sequence which performs the given [operation] from left to right on the current accumulator value and each element from left to right when the index of an element, current accumulator value and the element itself
 * and calculates the next accumulator value.
 * The operation is _terminal_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.reduceIndexedOrNull
 */
@SinceKotlin("1.4")
public inline fun <S, T : S> Sequence<T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S?): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++, accumulator, iterator.next()))
        index = index + 1
    }
    return accumulator
}
samples.collections.Collections.Aggregates.reduceOrNull

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <S, T : S>
    Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    var accumulator = operation(accumulator, iterator.next())
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}
```

```kotlin
/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * * Note that `acc` value passed to [operation] function should not be mutated;
 * * otherwise it would affect the previous value in resulting sequence.
 * * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later
 * because of sequence's lazy nature.
 * * The operation is _intermediate_ and _stateless_.
 * * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.runningFold(initial: R, operation: (acc: R, T) -> R):
    Sequence<R> {
        yield(initial)
        var accumulator = initial
        for (element in this@runningFold) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
```

```kotlin
/**
 * Returns a sequence containing successive accumulation values generated by applying [operation]
 * from left to right
 * to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.
 * * Note that `acc` value passed to [operation] function should not be mutated;
 * * otherwise it would affect the previous value in resulting sequence.
 * * The operation is _intermediate_ and _stateless_.
 * * @param [operation] function that takes the index of an element, current accumulator value
 * * and the element itself, and calculates the next accumulator value.
 * * @sample samples.Collections.Collections.Aggregates.runningReduceIndexed
 */
@SinceKotlin("1.4")
public fun <S, T : S> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S):
    Sequence<S> {
        yield(accumulator)
        var accumulator = operation(accumulator, iterator.next())
        while (iterator.hasNext()) {
            accumulator = operation(accumulator, iterator.next())
            yield(accumulator)
        }
    }
```

```kotlin
/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.
 * * Note that `acc` value passed to [operation] function should not be mutated;
 * * otherwise it would affect the previous value in resulting sequence.
 * * The operation is _intermediate_ and _stateless_.
 * * @param [operation] function that takes the index of an element, current accumulator value
 * * and the element itself, and calculates the next accumulator value.
 * * @sample samples.Collections.Collections.Aggregates.runningReduceIndexed
 */
@SinceKotlin("1.4")
public fun <S, T : S> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S):
    Sequence<S> {
        var accumulator = operation(accumulator, iterator.next())
        while (iterator.hasNext()) {
            accumulator = operation(accumulator, iterator.next())
            yield(accumulator)
        }
    }
```
yield(accumulator)
    return runningFold(initial, operation)
}
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting sequence.
 * The [initial] value should also be immutable (or should not be mutated)
 * as it may be passed to [operation] function later because of sequence's lazy nature.
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original sequence and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting sequence.
 * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * The operation is _intermediate_ and _stateless_.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * The operation is _intermediate_ and _stateless_.
 * @param samples.collections.Collections.Aggregates.scan
 *
 @SinceKotlin("1.4")
 @WasExperimental(ExperimentalStdlibApi::class)
 @public fun <T, R>
 Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return runningFold(initial, operation)
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun
```kotlin
<T> Sequence<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal
.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal
.InlineOnly
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.
 * The operation is _terminal_.
 */
public fun <T : Any> Sequence<T?>.requireNoNulls(): Sequence<T> {
    return map { it ?: throw IllegalArgumentException("null element found in $this.") }
}

/**
 * Splits this sequence into a sequence of lists each not exceeding the given [size].
 * The last list in the resulting sequence may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this sequence into several lists each not exceeding the given [size] and applies the given [transform] function to an each.
 * The last list may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.
 * @param transform function to apply to each list.
 * The operation is _intermediate_ and _stateful_.
 */
@SinceKotlin("1.2")
public fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements].
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            var removed = false
            return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements].
 * Note that the source sequence and the array being subtracted are iterated only when an `iterator` is requested from
 * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.
 * The elements array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
 * a correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {
    if (elements.isEmpty()) return this
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            return this@minus.filterNot { it in other }.iterator().iterator()
        }
    }
}
```

Open Source Used In DNAC Platform Arica  7963
operation, thus the elements are required to have
 * a correct and stable implementation of 'hashCode()' that
 doesn't change between successive invocations.
 * The operation is _intermediate_ and _stateful_.

 operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {
     return object: Sequence<T> {
         override fun iterator(): Iterator<T> {
             val other = elements.convertToSetForSetOperation()
             if (other.isEmpty())
                 return this@minus.iterator()
             else
                 return this@minus.filterNot { it in other }.iterator()
         }
     }
}

 /**
 * Returns a sequence containing all elements of original sequence
 * except the elements contained in the given [elements] sequence.
 * Note that the source sequence and the sequence being subtracted are iterated only when an `iterator` is requested from it.
 * The elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

 /**
 * Returns a sequence containing all elements of the original sequence
 * without the first occurrence of the given [element].
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> {
    return minus(element)
}

 /**
 * Splits the original sequence into pair of lists,
 * where *first* list contains elements for which [predicate] yielded `true`,
 * while *second* list contains elements for which [predicate] yielded `false`.
 * The operation is _terminal_ and _stateless_.
 */
public inline fun <T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    val first = ArrayList<T>()
    val second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

 /**
 * Returns a sequence containing all elements of the original sequence and then the given [element].
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.plus(element: T): Sequence<T> {
    return sequenceOf(this, sequenceOf(element)).flatten()
}

 /**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] array.
 * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from it.
 * Changing any of them between successive calls to `iterator` may affect the result.
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.plus(elements: Array<out T>): Sequence<T> {
    return this.plus(elements.asList())
}

 /**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] collection.
 * Note that the source sequence and the collection being added are iterated only when an `iterator` is requested from it.
 * Changing any of them between successive calls to `iterator` may affect the result.
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.plus(elements: Iterable<T>): Sequence<T> {
    return sequenceOf(this, elements.asSequence()).flatten()
}

 /**
 * Returns a sequence containing all elements of the original sequence and then the given [element].
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {
    return plus(element)
}
elements in this sequence.

* @param size the number of elements to take in each window
* @param step the number of elements to move the window forward by on each step, by default 1
* @param partialWindows controls whether or not to keep partial windows in the end if any
* by default `false` which means partial windows won't be preserved
* @param sample samples.collections.Sequences.Transformations.takeWindows
* @SinceKotlin("1.2")

* Returns a sequence containing the results of applying the given [transform] function to each pair of adjacent elements.
* The returned sequence is empty if this sequence contains less than two elements.
* The operation is _intermediate_ and _stateless_.
* @sample samples.collections.Sequences.Transformations.zipWithNext
* @SinceKotlin("1.2")

    public fun <T, R> Sequence<T>.zipWithNext(transform: (a: T, b: T) -> R): Sequence<R> { return sequence { val iterator = iterator()
* while (iterator.hasNext()) {
* yield(transform(iterator.next())
* current = next
* } yield(transform(current, next))
* } yield(transform(current, next))
* }
* Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
* If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo
* @public fun <T, A : Appendable> Sequence<T>.joinTo(buffer: A, separator: CharSequence = "", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A
* {
* var current = 0
* for (element in this) {
* if (++count > 1)
buffer.append(separator)
 if (limit < 0 || count <= limit) {
     buffer.appendElement(element, transform)
 } else break
 }
 if (limit > 0 && count > limit) buffer.append(truncated)
 buffer.append(postfix)
 return buffer
}

/**
 * Creates a string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * The operation is _terminal_.
 * @sample
 * samples.collections.Collections.transform.joinTo(StringBuilder, separator, prefix, postfix, limit, truncated, transform).toString()
 */
public fun <T>
    Sequence<T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.
 * @sample
 * Sequence<T>.asIterable().toList()
 */
public fun <T>
    Sequence<T>.asIterable(): Iterable<T> {
    return Iterable { this.iterator() }
}

/**
 * Returns this sequence as a [Sequence].
 * @sample
 * Sequence<Int>.asSequence().sum()
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.asSequence(): Sequence<T> {
    return this
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Sequence<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Sequence<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Sequence<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Sequence<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Sequence<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Sequence<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Sequence<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Sequence<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Sequence<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Sequence<Long>.sum(): Long {
    var sum: Long = 0
    for (element in this) {
        sum += element
    }
    return sum
}
for (element in this) {\n    sum += element\n} \nreturn sum\n
* Returns the sum of all elements in the sequence.\n* The operation is _terminal_.\n* @param @jvm.NameValue("sumOfDouble")\n* @public\n
func Sequence<Double>.sum(): Double {\n    var sum: Double = 0.0\n    for (element in this) {\n        sum += element\n    } \nreturn sum\n
*/ Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*kotlin.collections*\n
NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib/kotlin/nimprot kotlin.random.\n*kotlin.ranges*\n
*/Returns a set containing all elements of the original set except the given [element].\n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.minus(element: T): Set<T> {\n    var result = LinkedHashSet<T>(mapCapacity(size))\n    var removed = false\n    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }\n}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] array.\n* The returned set preserves the element iteration order of the original set.\n* The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have\n* a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.\n*\n*public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {\n    var result = LinkedHashSet<T>(this)\n    result.removeAll(elements)\n    return result\n}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.\n* The returned set preserves the element iteration order of the original set.\n* The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have\n* a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.\n*\n*public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {\n    var result = LinkedHashSet<T>(this)\n    result.removeAll(elements)\n    return result\n}

/**
 * Returns a set containing all elements of the original set except the given [elements] sequence.\n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {\n    var result = LinkedHashSet<T>(this)\n    result.removeAll(elements)\n    return result\n}

/**
 * Returns a set containing all elements of the original set except the given [elements] \n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.minusElement(element: T): Set<T> {\n    var result = LinkedHashSet<T>(this)\n    result.remove(element)\n    return result\n}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.\n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.plus(element: T): Set<T> {\n    var result = LinkedHashSet<T>(mapCapacity(size + 1))\n    result.addAll(this)\n    result.add(element)\n    return result\n}

/**
 * Returns a set containing all elements of the original set and the given [elements], which aren't already in this set.\n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {\n    var result = LinkedHashSet<T>(mapCapacity(this.size + elements.size))\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}

/**
 * Returns a set containing all elements of the original set and the given [elements], which aren't already in this set.\n* The returned set preserves the element iteration order of the original set.\n*\n*public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {\n    var result = LinkedHashSet<T>(mapCapacity(elements.collectionSizeOrNull()!! + this.size + 1))\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}

}
returned set preserves the element iteration order of the original set.

\[
\text{Set<T>.plus(elements: Sequence<T>): Set<T> \{ val result = LinkedHashSet<T>(mapCapacity(this.size * 2))
\text{result.addAll(this)}
\text{result.addAll(elements)}
\text{return result}\}
\]

\*\*

\text{Set<T>.plus(element: T): Set<T> \{ return plus(element)\}}

\* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

\*\*

\*\*

\*\*
for (index in indices.reversed()) { if (predicate(this[index])) { return index } } return -1

public fun CharSequence.last(): Char { if (isEmpty()) throw NoSuchElementException("Char sequence is empty.") return this[lastIndex]}

/**
 * Returns the last character matching the given [predicate].
 * @throws [NoSuchElementException] if no such character is found.
 */
public inline fun CharSequence.last(predicate: (Char) -> Boolean): Char? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random character from this char sequence.
 * @throws [NoSuchElementException] if this char sequence is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharSequence.randomOrNull(): Char? {
    if (isEmpty()) return null
    return get(random.nextInt(length))
}

/**
 * Returns a subsequence of this char sequence with the first [n] characters removed.
 * @throws IllegalArgument Exception if [n] is negative.
 */
public fun CharSequence.drop(n: Int): CharSequence {
    if (n < 0) throw IllegalArgumentException("Requested character count \$n is less than zero.")
    return subSequence(n.coerceAtMost(length), length)
}

/**
 * Returns a string with the first [n] characters removed.
 * @throws IllegalArgument Exception if [n] is negative.
 */
public fun String.drop(n: Int): String {
    if (n < 0) throw IllegalArgumentException("Requested character count \$n is less than zero.")
    return substring(n.coerceAtMost(length))
}
this char sequence with the last [n] characters removed.

* @throws IllegalArgumentException if [n] is negative.

* @sample samples.text.Strings.drop
*/

public fun CharSequence.dropLast(n: Int):
        String {
            require(n >= 0) { "Requested character count $n is less than zero." }
            return take((length - n).coerceAtLeast(0))
        }

/**
 * Returns a string containing all characters except last characters that satisfy the given [predicate].
 * 
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * 
 * @param [destination] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * 
 * @sample samples.collections.Collections.Filtering.filterIndexedTo
 */

public inline fun <C : Appendable>
        CharSequence.filterIndexedTo(destination: C, predicate: (Int, Char) -> Boolean): C {
            for (index in this.indices)
                if (!predicate(index, this[index]))
                    destination.append(this[index])
            return destination
        }

/**
 * Returns a char sequence containing only those characters from the original char sequence that do not match the given [predicate].
 * 
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * 
 * @sample samples.collections.Collections.Filtering.filterIndexed
 */

public inline fun <C : Appendable>
        CharSequence.filterIndexedTo(destination: C, predicate: (Int, Char) -> Boolean): C {
            for (index in this.indices)
                if (predicate(index, this[index]))
                    destination.append(this[index])
            return destination
        }

/**
 * Returns a string containing only those characters from the original string that match the given [predicate].
 * 
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * 
 * @sample samples.collections.Collections.Filtering.filterTo
 */

public inline fun <C : Appendable>
        String.filterTo(destination: C, predicate: (Int, Char) -> Boolean): C {
            for (index in this.indices)
                if (predicate(index, this[index]))
                    destination.append(this[index])
            return destination
        }

/**
 * Returns a string containing only those characters from the original string that do not match the given [predicate].
 * 
 * @param [predicate] function that takes the index of a character and the character itself
 * and returns the result of predicate evaluation on the character.
 * 
 * @sample samples.collections.Collections.Filtering.filterTo
 */

public inline fun <C : Appendable>
        String.filterTo(destination: C, predicate: (Int, Char) -> Boolean): C {
            for (index in this.indices)
                if (predicate(index, this[index]))
                    destination.append(this[index])
            return destination
        }
Appendable> CharSequence.filterNotTo(destination: C, predicate: (Char) -> Boolean): C { \n    for (element in this) \n        if (!predicate(element)) destination.append(element) \n    return destination \n} \n
/**
 * Appends all characters matching the given [predicate] to the given [destination].\n * @param samples\n * @param collections.Collections.Filtering.filterTo
 */
public inline fun <C : Appendable> CharSequence.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this)
        if (!predicate(element)) destination.append(element)
    return destination
}

/**
 * Returns a char sequence containing characters of the original char sequence at the specified range of [indices].\n */
public fun CharSequence.slice(indices: IntRange): CharSequence {
    if (indices.isEmpty()) return ""
    return subSequence(indices)
}

/**
 * Returns a string containing characters of the original string at specified [indices].\n */
@kotlin.internal.InlineOnly
public inline fun String.slice(indices: Iterable<Int>): String {
    return (this as CharSequence).slice(indices).toString()
}

/**
 * Returns a subsequence of this char sequence containing the first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n */
public fun CharSequence.take(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(0, n.coerceAtMost(length))
}

/**
 * Returns a string containing characters of the original string at specified [indices].\n */
public fun CharSequence.reversed(): CharSequence {
    return
}

/**
 * Returns a subsequence of this char sequence containing the last [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.\n */
public fun CharSequence.reversed(): CharSequence {
    return
}

/**
 * Returns a string containing last characters that satisfy the given [predicate].\n */
public inline fun CharSequence.takeLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return subSequence(index + 1, length)
        }
    return subSequence(0, length)
}

/**
 * Returns a char sequence containing characters of the reversed order.\n */
public inline fun CharSequence.reversed(): CharSequence {
    return
}

Open Source Used In DNAC Platform Arica  7971
StringBuilder(this).reverse() Returns a string with characters in reversed order.

/**
 * Returns a string with characters in reversed order.
 */
@kotlin.internal.InlineOnly
public inline fun String.reversed(): String {
    return (this as CharSequence).reversed().toString()
}

/**
 * Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to characters of the given char sequence.
 * If any two of pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original char sequence.
 * @sample samples.text.Strings.associate
 */
public inline fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by
 * [keySelector] functions applied to characters of the given char sequence.
 * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original char sequence.
 * @sample samples.text.Strings.associateWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> CharSequence.associateTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

Open Source Used In DNAC Platform Arica
former value in the map.

/*
 * @sample samples.text.Strings.associateWithTo
 */

@SinceKotlin("1.3")
public inline fun <V, M : MutableMap<Char, V>>
CharSequence.associateWithTo(destination: M, valueSelector: (Char) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

*/

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <<C : MutableCollection<in Char>>
CharSequence.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <C : MutableCollection<in Char>>
CharSequence.toMutableList(): C {
    return toCollection(ArrayList<Char>()).toCollection(Cсон())
}

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <C : MutableCollection<in Char>>
CharSequence.toSet(): C {
    return toCollection(LinkedHashSet<Char>(mapCapacity(length.coerceAtMost(128)))).toCollection(Cсон())
}

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <R> CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <R, C : MutableCollection<in R>>
CharSequence.flatMapIndexedTo(destination: C, transform: (index: Int, Char) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++,
        element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all characters to the given [destination] collection.
 */
@SinceKotlin("1.3")
public inline fun <R, C : MutableCollection<in R>>
CharSequence.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups characters of the original char sequence by the key returned by the given [keySelector] function
 * applied to each character and its index in the original char sequence, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedIterable")
@kotlin.internal.InlineOnly
public inline fun <K, V>
Charsquence.groupBy((keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups characters of the original char sequence by the key returned by the given [keySelector] function
 * by the key returned by the given [keySelector] function applied to each character and puts to the [destination] map each group key associated with a list of corresponding characters.
 * @return The [destination] map.
 */
public inline fun <K, M : MutableMap<K, MutableList<V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K): M {
    var retval = destination
    for (element in this) {
        val key = keySelector(element)
        val list = retval.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return retval
}

/**
 * Creates a [Grouping] source from a char sequence to be used later with one of group-and-fold operations
 * by using the specified [keySelector] function to extract a key from each character.
 * @return The [Grouping] source.
 */
@SinceKotlin("1.1")
public inline fun <K> CharSequence.groupingBy(crossinline keySelector: (Char) -> K): Grouping<Char, K> {
    return object :
        Grouping<Char, K> {
            override fun sourceIterator(): Iterator<Char> = this@groupingBy.iterator()
            override fun keyOf(element: Char): K = keySelector(element)
        }
}

/**
 * Returns a list containing the results of applying the given [transform] function to each character in the original char sequence.
 * @param [transform] function that takes the index of a character and the character itself and returns the result of the transform applied to the character.
 * @return The list.
 */
public inline fun <R> CharSequence.map(transform: (Char) -> R): List<R> {
    return mapTo(ArrayList<R>(length), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each character and its index in the original char sequence.
 * @param [transform] function that takes the index of a character and the character itself and returns the result of the transform applied to the character.
 * @return The list.
 */
public inline fun <R : Any> CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

/**
 * Applies the given [transform] function to each character and its index in the original char sequence
 * and appends only the non-null results to the given [destination].
 * @param [transform] function that takes the index of a character and the character itself and returns the result of the transform applied to the character.
 * @return The list.
 */
public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each character and its index in the original char sequence
 * and appends only the non-null results to the given [destination].
 * @param [transform] function that takes the index of a character and the character itself and returns the result of the transform applied to the character.
 * @return The list.
 */
public inline fun <R : Any> CharSequence.mapIndexedNotNullTo(destination: R): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}
and appends only the non-null results to the given [destination].\n */
public inline fun <R : Any, C : MutableCollection<R>> CharSequence.mapTo(destination: C, transform: (Char) -> R?): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each character of the original char sequence
 * and appends the results to the given [destination].\n * Returns a lazy [Iterable] that wraps each character of the original char sequence into an
 * [IndexedValue] containing the index of that character and the character itself.\n */
public fun CharSequence.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns `true` if all characters match the given [predicate].\n */
public inline fun CharSequence.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if char sequence has at least one character.\n */
public fun CharSequence.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if at least one character matches the given [predicate].\n */
public inline fun CharSequence.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the length of this char sequence.\n */
@kotlin.internal.InlineOnly
public inline fun CharSequence.count(): Int {
    return length
}

/**
 * Returns the number of characters matching the given [predicate].\n */
public inline fun CharSequence.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with \[initial\] and applying \[operation\] from left to right
 * to each character and current accumulator value.\n * Returns the specified \[initial\] value if the char
 * sequence is empty.\n */
public inline fun <R> CharSequence.fold(initial: R, operation: (R, Char) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with \[initial\] and applying \[operation\] from right to left
 * to each character and current accumulator value.\n * Returns the specified \[initial\] value if the char
 * sequence is empty.\n */
public inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, R) -> R): R {
    var index = 0
    var accumulator = initial
    while (index < length) {
        accumulator = operation(get(index++), accumulator)
        ++index
    }
    return accumulator
}

/**
 * Performs the given \[action\] on each character.\n */
public inline fun CharSequence.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}
the action on the character.
 */
public inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit):
    Unit { 
        var index = 0
        for (item in this) action(index++, item)
    }

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.max(): Char? {
    return maxOrNull()
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharSequence.maxBy(selector: (Char) -> R): Char? {
    var maxElem = this[0]
    var lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxElem = this[i]
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each character in the char sequence.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 * 
 * @since 1.4
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.maxOf(selector: (Char) -> R): R {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each character in the char sequence or `null` if there are no
 * characters.
 * 
 * @throws NoSuchElementException if the char sequence is empty.
 * 
 * @since 1.4
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R? {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
public inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxVal < v) {
            maxVal = v
        }
    }
    return maxVal
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxVal < v) {
            maxVal = v
        }
    }
    return maxVal
}

/**
 * Returns the largest character or 'null' if there are no characters.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public fun CharSequence.maxOrNull(): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.", ReplaceWith("maxWithOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.maxWithOrNull(selector: (Char) -> R): Char? {
    return maxWithOrNull(selector)
}

/**
 * Returns the first character having the largest value according to the provided [comparator] or 'null' if there are no characters.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public fun CharSequence.maxWithOrNull(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.min(): Char? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minVal > v) {
            minVal = v
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minVal < v) {
            minVal = v
        }
    }
    return minVal
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public fun CharSequence.minOrNull(): R? {
    if (isEmpty()) return null
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minVal > v) {
            minVal = v
        }
    }
    return minVal
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(selector: (Char) -> R): R? {
    return minWithOrNull(selector)
}

/**
 * Returns the first character yielding the smallest value of the given function or 'null' if there are no characters.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public fun CharSequence.minWithOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        val e = this[i]
        if (minVal > e) minVal = e
    }
    return minVal
}

/**
 * Returns the smallest character or 'null' if there are no characters.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public fun CharSequence.minOrNull(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.minWithOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each character in the char sequence.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.minOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minVal < v) {
            minVal = v
        }
    }
    return minVal
}
result is `NaN`.\n
*\n* \n* \n* @throws NoSuchElementException if the char sequence is empty.\n
*\n* @SinceKotlin("1.4")\n* @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n* @OverloadResolution ByLambdaReturnType\n* @kotlin.internal.InlineOnly\n* @public inline fun CharSequence.minOf(selector: (Char) -> Float): Float {\n  \n  if (isEmpty()) throw NoSuchElementException()\n  \n  var minValue = selector(this[0]) for (i in 1..lastIndex) {\n    val v = selector(this[i])\n    minValue = minOf(minValue, v)\n  }\n  \n  return minValue\n} \n
/**\n * Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value according to the provided [comparator] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest character or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value according to the provided [comparator] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest value among all values produced by [selector] function \n* applied to each character in the char sequence or `null` if there are no characters.\n* Returns the smallest character or `null` if there are no characters.
    val e = this[i]
    if (min > e) min = e
  }
  return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin("1.4")@Published func CharSequence.minWith(@comparator: Comparator<in Char>): Char? {
  var min = this[0]
  for (i in 1..lastIndex) {
    val e = this[i]
    if (@comparator.compare(min, e) > 0) min = e
  }
  return min
}

/**
 * Returns the first character having the smallest value according to the
 * provided [comparator] or `null` if there are no characters.
 */
@SinceKotlin("1.4")@Published func CharSequence.minWithOrNull(@comparator: Comparator<in Char>): Boolean {
  return isEmpty() || @comparator.compare(min, this[0]) > 0
}

/**
 * Returns `true` if the char sequence has no characters.
 */
public fun CharSequence.none(): Boolean {
  return isEmpty()
}

/**
 * Returns `true` if no characters match the given [predicate].
 */
public inline fun CharSequence.none(predicate: (Char) -> Boolean): Boolean {
  for (element in this) if (predicate(element)) return false
  return true
}

/**
 * Performs the given [action] on each character and returns the char sequence itself.
 */
@SinceKotlin("1.1") public inline fun <S : CharSequence> S.onEach(action: (Char) -> Unit): S {
  return apply { for (element in this) action(element) }
}

/**
 * Performs the given [action] on each character, providing sequential index with the character,
 * and returns the char sequence itself.
 */
@SinceKotlin("1.4") public inline fun <S : CharSequence> S.onEachIndexed(action: (index: Int, Char) -> Unit): S {
  return apply { forEachIndexed(action) }
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to right
 * to current accumulator value and each character.
 */
public inline fun CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char {
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(accumulator, this[index])
  }
  return accumulator
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to right
 * to current accumulator value and each character with its index in the original char sequence.
 */
public inline fun CharSequence.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to right
 * to current accumulator value and each character with its index in the original char sequence.
 */
public inline fun CharSequence.reduceIndexedOrNull(operation: (index: Int, acc: Char, Char) -> Char): Char? {
  var accumulator = this[0]
  for (index in 1..lastIndex) {
    accumulator = operation(index, accumulator, this[index])
  }
  return accumulator
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to right
 * to current accumulator value and each character.
 */
public inline fun CharSequence.reduceOrNull(operation: (acc: Char) -> Unit): Char? {
  return apply { for (element in this) action(element) }
}
CharSequence.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? { if (isNotEmpty()) return null

var accumulator = this[0] for (index in 1..lastIndex) { accumulator = operation(accumulator, this[index])

return accumulator
}

accumulate value starting with the last character and applying [operation] from right to left

* @param [operation] function that takes a character and current accumulator value,
* @param samples.Collections.Aggregates reduceOrNull

CharSequence.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? { if (isNotEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")

var accumulator = get(index--) while (index >= 0) { accumulator = operation(get(index--), accumulator)

return accumulator
}

accumulate value starting with the last character and applying [operation] from right to left

* @param [operation] function that takes the index of a character, the character itself and current accumulator value,
* @param sample

CharSequence.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? { if (isEmpty()) return null

var accumulator = get(index--) while (index >= 0) { accumulator = operation(index, get(index--), accumulator)

return accumulator
}

accumulates value starting with the last character and applying [operation] from right to left

* @param [operation] function that takes the index of a character, the character itself and current accumulator value,
* @param sample

CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char? { if (isEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")

var accumulator = get(index--) while (index >= 0) { accumulator = operation(index, get(index--), accumulator)

return accumulator
}

- @param [operation] function that takes the index of a character, the character itself and current accumulator value,
* @param sample

CharSequence.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char? { if (isEmpty()) return null

var accumulator = get(index--) while (index >= 0) { accumulator = operation(index, get(index--), accumulator)

return accumulator
}

accumulate value starting with the last character and applying [operation] from right to left

* @param [operation] function that takes a character and current accumulator value,
* @param sample

CharSequence.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? { if (isNotEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")

var accumulator = get(index--) while (index >= 0) { accumulator = operation(get(index--), accumulator)

return accumulator
}

accumulate value starting with the last character and applying [operation] from right to left

* @param [operation] function that takes a character and current accumulator value,
* @param sample

CharSequence.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> { if (isNotEmpty()) return emptyList()

val result = ArrayList<R>(length + 1) apply { add(initial) }

var accumulator = initial while (index >= 0) { accumulator = operation(accumulator, element)

return result.accumulate value starting with the initial value

* @param [operation] function that takes a character and current accumulator value,
* @param sample

CharSequence.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> { if (isNotEmpty()) return emptyList()

val result = ArrayList<R>(length + 1) apply { add(initial) }

var accumulator = initial while (index >= 0) { accumulator = operation(accumulator, element)

return result.accumulate value starting with the initial value

* @param [operation] function that takes a character and current accumulator value,
* @param sample
```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying 
 * [operation] from left to right\n to each character and current accumulator value 
 * that starts with the first character of this char sequence.\n *\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n *\n * @param \[operation\] function that takes current accumulator value and a character, 
 * and calculates the next accumulator value.\n *\n * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <R> CharSequence.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
    return runningFold(initial, operation)
}

@Deprecated("Use runningReduceIndexed instead.",
            ReplaceWith("runningReduceIndexed(operation)"), level =
            DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <R> CharSequence.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}
```

DeprecationLevel.ERROR)\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic inline fun
CharSequence.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {\n    return runningReduceIndexed(operation)\n}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun
CharSequence.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfDouble")\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.sumOf(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfInt")\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.sumOf(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfLong")\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.sumOf(selector: (Char) -> Long): Long {
    var sum: Long = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfUInt")\n@kotlin.experimental.UnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.sumOf(selector: (Char) -> UInt): UInt {
    var sum: UInt = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@kotlin.experimental.UnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
CharSequence.sumOf(selector: (Char) -> ULong): ULong {
    var sum: ULong = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Splits this char sequence into a list of strings each not exceeding the given [size].
 * The last string in the resulting list may have less characters than the given [size].
 */
@sample samples.text.Strings.chunked
@SinceKotlin("1.2")\n@public fun CharSequence.chunked(size: Int): List<String> {\n    return windowed(size, size, partialWindows = true)\n}

/**
 * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to each.
 * The last string in the resulting list may have less characters than the given [size].
 */
@sample samples.text.Strings.chunkedTransform
@SinceKotlin("1.2")\n@public fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {\n    return windowed(size, size, partialWindows = true, transform = transform)\n}

/**
 * Splits this char sequence into a sequence of strings each not exceeding the given [size] and applies the given [transform] function to each.
 * The last string in the resulting list may have less characters than the given [size].
 */
@sample samples.text.Strings.chunked
@SinceKotlin("1.2")\n@public fun CharSequence.chunked(size: Int): List<String> {\n    return windowed(size, size, partialWindows = true)\n}
number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.

```kotlin
public fun CharSequence.chunkedSequence(size: Int): Sequence<String> {
    return chunkedSequence(size) { it.toString() }
}
```

Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each. You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last char sequence may have less characters than the given [size].

```kotlin
@SinceKotlin("1.2")
public fun CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {
    return windowedSequence(size, transform = transform) { it.toString() }
}
```

Splits the original char sequence into pair of char sequences, where *first* char sequence contains characters for which [predicate] yielded `true`, while *second* char sequence contains characters for which [predicate] yielded `false`. You should not store it or allow it to escape in some way, unless you made a snapshot of it.

```kotlin
@SinceKotlin("1.2")
public fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R), windowedSequence(size, partialWindows = true) { it.toString() }
```

Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

```kotlin
@SinceKotlin("1.2")
public fun CharSequence.chunkedTransformToSequence(): Sequence<String> {
    return windowedSequence(size, transform = transform) { it.toString() }
}
```

Splits this char sequence into several sequences each not exceeding the given [size] and applies the given [transform] function to an each. You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last sequence may have less characters than the given [size].

```kotlin
@SinceKotlin("1.2")
public fun <R> CharSequence.chunkedTransformToSequence(size: Int, partialWindows: Boolean = false, transform: (CharSequence) -> R) {
    return windowedSequence(size, transform = transform) { it.toString() }
}
```

Splits the original char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each. You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last char sequence may have less characters than the given [size].

```kotlin
@SinceKotlin("1.2")
public fun <R> CharSequence.chunkedTransformToSequence(size: Int, transform: (CharSequence) -> R) {
    return windowedSequence(size, transform = transform) { it.toString() }
}
```

Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

```kotlin
@SinceKotlin("1.2")
public fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first, second)
}
```

Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

```kotlin
@SinceKotlin("1.2")
public fun <R> CharSequence.partition(predicate: (Char) -> Boolean): Pair<String, String> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first.toString(), second.toString())
}
```

Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

```kotlin
@SinceKotlin("1.2")
public fun CharSequence.takeWindows(): List<String> {
    return takeWindows() { it.toString() }
}
```

Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded `true`, while *second* string contains characters for which [predicate] yielded `false`. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.
Open Source Used In DNAC Platform Arica 7984

```kotlin
/**
 * Returns a sequence of snapshots of the window of the given [size]\n * sliding along this char sequence with the given [step], where each
 * snapshot is a string.\n * Several last strings may have less characters than the given [size].\n * Both [size] and [step] must be positive and can be
 * greater than the number of elements in this char sequence.\n * @param step the number of elements to take in each window\n * @param size the number of elements to move the window forward by on an each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.takeWindows\n */
@SinceKotlin("1.2")
public fun CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {
    (it.toString() \n    return windowedSequence(size, step, partialWindows) { it.toString() \n
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step
    return windows.asSequence().map { index ->
        val end = index + size
        val coercedEnd = if (end < 0 || end > length) length else end
        transform(subSequence(index, coercedEnd))
    }
}

/**
 * Returns a sequence of results
 * of applying the given [transform] function to
 * an each char sequence representing a view over the window of the given [size]\n * sliding along this char sequence with the given [step].\n * Note that the char sequence passed to
 * the [transform] function is ephemeral and is valid only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot of it.\n * Several last char sequences may have less characters than the given [size].\n * Both [size] and [step] must be positive and can be greater than the number of elements in this
 * char sequence.\n * \n * @param size the number of elements to take in each window\n * \n * @param step the number of elements to move the window forward by on an each step, by default 1\n * \n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n * \n * @sample samples.collections.Sequences.Transformations.averageWindows\n */
@SinceKotlin("1.2")
public fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): Sequence<R> {
    checkWindowSizeStep(size, step)
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step
    return windows.asSequence().map { index ->
        val end = index + size
        val coercedEnd = if (end < 0 || end > length) length else end
        transform(subSequence(index, coercedEnd))
    }
}

/**
 * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index\n * The returned list has length of the shortest char sequence.\n * \n * @sample samples.text.Strings.zip\n */
public infix fun CharSequence.zip(other: CharSequence): List<Pair<Char, Char>> {
    return zip(other) { c1, c2 -> c1 to c2 }
}

/**
 * Returns a list of values built from the characters of `this` and the [other] char sequences with the
 * provided [transform] function applied to each pair of characters.\n * \n * @sample samples.text.Strings.zipWithTransform\n */
public inline fun <V> CharSequence.zip(other: CharSequence, transform: (a: Char, b: Char) -> V): List<V> {
    val length = minOf(this.length, other.length)
    val list = ArrayList<V>(length)
    for (i in 0 until length) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to an each pair of two adjacent characters in this char sequence.\n * \n * @sample samples.collections.Collections.Transformations.zipWithNext\n */
@SinceKotlin("1.2")
public inline fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {
    return zipWithNext(transform) \n}

/**
 * Creates an [Iterable] instance
 * that wraps the original char sequence returning its characters when being iterated.\n * \n * @sample samples.collections.Collections.Transforms.zipWithNextToFindDeltas\n */
@SinceKotlin("1.2")
public fun <R> CharSequence.zipWithNextToFindDeltas(): InclusiveSequence<R> \n}

Open Source Used In DNAC Platform Arica 7984
```
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

import kotlin.contracts.contract

/**
 * Returns a sub sequence of this char sequence having leading and trailing characters matching the [predicate] removed.
 */
public inline fun CharSequence.trim(predicate: (Char) -> Boolean): CharSequence {
    var startIndex = 0
    var endIndex = length
    var startFound = false

    while (startIndex <= endIndex) {
        val index = if (!startFound) startIndex else endIndex
        val match = predicate(this[index])

        if (!startFound) {
            if (!match)
                startFound = true
            else
                startIndex += 1
        } else {
            if (!match)
                break
            else
                endIndex -= 1
        }
    }

    return subSequence(startIndex, endIndex + 1)
}

/**
 * Returns a string having leading and trailing characters matching the [predicate] removed.
 */
public inline fun String.trim(predicate: (Char) -> Boolean): String =
    this as CharSequence.trim(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading characters matching the [predicate] removed.
 */
public inline fun CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)

    return ""
}

/**
 * Returns a string having leading characters matching the [predicate] removed.
 */
public inline fun String.trimStart(predicate: (Char) -> Boolean): String =
    this as CharSequence.trimStart(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.
 */
public inline fun CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices.reversed())
        if (!predicate(this[index]))
            return subSequence(0, index + 1)

    return ""
}

/**
 * Returns a string having trailing characters matching the [predicate] removed.
 */
public inline fun String.trimEnd(predicate: (Char) -> Boolean): String =
    this as CharSequence.trimEnd(predicate).toString()

/**
 * Returns a char sequence with content of this char sequence padded at
 * the beginning by the specified [length] with the specified character or space.
 */
@sample
public fun String.padStart(@param length: Int, @param padChar: Char): String =
    this + String(length, padChar)

/**
 * Returns a char sequence with content of this char sequence padded at
 * the end by the specified [length] with the specified character or space.
 */
@sample
public fun String.padEnd(@param length: Int, @param padChar: Char): String =
    String(length, padChar) + this

/**
 * Returns a char sequence padded at the beginning and end by the specified [length] with the specified character.
 */
@sample
public fun String.padBoth(@param length: Int, @param padChar: Char): String =
    padStart(length, padChar) + padEnd(length, padChar)

/**
 * Returns a char sequence having leading and trailing whitespace removed.
 */
public fun String.trim(): String =
    this as String.trim().toString()

/**
 * Returns a char sequence having leading whitespace removed.
 */
@kotlin.internal.InlineOnly
public inline fun String.trimStart(): String =
    this as String.trimStart().toString()

/**
 * Returns a char sequence having trailing whitespace removed.
 */
@kotlin.internal.InlineOnly
public inline fun String.trimEnd(): String =
    this as String.trimEnd().toString()

/**
 * Returns a char sequence having leading and trailing whitespace removed.
 */
public fun String.trimWhitespace(): String =
    this as String.trimWhitespace().toString()
samples.text.Strings.padStart

*private fun CharSequence.padStart(length: Int, padChar: Char = ' '):

CharSequence {
    if (length < 0) throw IllegalArgumentException("Desired length $length is less than zero.")

    if (length <= this.length) return this.subSequence(0, this.length)

    val sb = StringBuilder(length)
    for (i in 1..(length - this.length)) sb.append(padChar)
    sb.append(this)
    return sb
}

/**
 * Pads the string to the specified [length] at the beginning with the specified character or space.
 */
public fun CharSequence.padStart(length: Int, padChar: Char = ' '):

*private fun String.padStart(length: Int, padChar: Char = ' '):

*private fun String.padEnd(length: Int, padChar: Char = ' '):

*private fun CharSequence?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

/**
 * Returns `true` if this char sequence is empty (contains no characters).
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isEmpty(): Boolean = length == 0

/**
 * Returns `false` if this char sequence is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotEmpty(): Boolean = length > 0

/**
 * Returns `true` if this nullable character sequence is either `null` or empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}

/**
 * Iterator for characters of the given char sequence.
 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private var index = 0

    public override fun nextChar(): Char = get(index++)

    public override fun hasNext(): Boolean = index < length
}

/** Returns the string if it is not `null`, or the empty string otherwise.
 */
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""

/**
 * Returns this character sequence if it's not empty
 * or the result of calling [defaultValue] function if the character sequence is empty.
 *
 * @sample samples.text.Strings.stringIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns this char sequence padded with [padChar] as many times as are necessary to reach that length.
 */
public fun CharSequence.padStart(length: Int, padChar: Char = ' '):

*private fun String.padStart(length: Int, padChar: Char = ' '):

*private fun String.padEnd(length: Int, padChar: Char = ' '):

*private fun CharSequence?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

/**
 * Returns `true` if this character sequence is empty (contains no characters).
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isEmpty(): Boolean = length == 0

/**
 * Returns `false` if this character sequence is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotEmpty(): Boolean = length > 0

/**
 * Returns `true` if this character sequence is not empty and contains some characters except of whitespace characters.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotBlank(): Boolean = !isBlank()

/**
 * Returns `true` if this nullable character sequence is either `null` or empty or consists solely of whitespace characters.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}

/**
 * Iterator for characters of the given char sequence.
 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private var index = 0

    public override fun nextChar(): Char = get(index++)

    public override fun hasNext(): Boolean = index < length
}

/** Returns the string if it is not `null`, or the empty string otherwise.
 */
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""

/**
 * Returns this character sequence if it's not empty
 * or the result of calling [defaultValue] function if the character sequence is empty.
 *
 * @sample samples.text.Strings.stringIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns `true` if this nullable character sequence is either `null` or empty or consists solely of whitespace characters.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}

/**
 * Iterator for characters of the given char sequence.
 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private var index = 0

    public override fun nextChar(): Char = get(index++)

    public override fun hasNext(): Boolean = index < length
}

/** Returns the string if it is not `null`, or the empty string otherwise.
 */
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""

/**
 * Returns this character sequence if it's not empty
 * or the result of calling [defaultValue] function if the character sequence is empty.
 *
 * @sample samples.text.Strings.stringIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns `true` if this nullable character sequence is either `null` or empty or consists solely of whitespace characters.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}

/**
 * Returns `true` if this character sequence is empty (contains no characters).
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isEmpty(): Boolean = length == 0

/**
 * Returns `false` if this character sequence is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotEmpty(): Boolean = length > 0

/**
 * Returns `true` if this character sequence is not empty and contains some characters except of whitespace characters.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotBlank(): Boolean = !isBlank()
sequence if it is not empty and doesn't consist solely of whitespace characters, * or the result of calling [defaultValue] function otherwise.

* @SinceKotlin("1.3")* @kotlin.internal.InlineOnly

** Returns the index of the last character in the char sequence or -1 if it is empty.

* @kotlin.internal.InlineOnly
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning@Deprecated("Use parameters named startindex and endIndex.")

* @param startindex the start index (inclusive).
* @param endIndex the end index (exclusive). If not specified, the length of the char sequence is used.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).

* @param startindex the start index (inclusive). * This extension is chosen only for invocation with old-named parameters.

* @param endindex the end index (exclusive).
original string.

```kotlin
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}
```

```kotlin
/**
 * Returns a substring after the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}
```

```kotlin
/**
 * Returns a substring after the last occurrence of [delimiter].
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}
```

```kotlin
/**
 * Returns a char sequence with content of this char sequence where its part at the given range is replaced with the [replacement] char sequence.
 * @param startIndex the index of the first character to be replaced.
 * @param endIndex the index of the first character after the replacement to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun CharSequence.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb
}

/**
 * Returns a char sequence with content of this char sequence where its part at the given range is replaced with the [replacement] char sequence.
 * @param startIndex the index of the first character to be replaced.
 * @param endIndex the index of the first character after the replacement to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun CharSequence.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb
}
```

```kotlin
/**
 * Replaces the part of the string at the given range with the [replacement] string.
 * @param startIndex the index of the first character to be removed.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@kotlin.internal.InlineOnly
public inline fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: String): String =
    (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()

/**
 * Replaces the part of the string at the given range with the [replacement] char sequence.
 * @param startIndex the index of the first character to be replaced.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@kotlin.internal.InlineOnly
public inline fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String =
    (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()
```

```kotlin
/**
 * Removes the part of a string at the given range.
 * @param startIndex the index of the first character to be removed.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun String.removeRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    if (endIndex <= startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb
}

/**
 * Removes the part of a string at the given range.
 * @param startIndex the index of the first character to be removed.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun String.removeRange(startIndex: Int, endIndex: Int, replacement: String): CharSequence {
    if (endIndex <= startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb
}
```

```kotlin
/**
 * Removes the part of a string at the given range.
 * @param startIndex the index of the first character to be removed.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun String.removeRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String =
    (this as CharSequence).removeRange(range).toString()

/**
 * Removes the part of a string at the given range.
 * @param startIndex the index of the first character to be removed.
 * @param endIndex the index of the first character after the removed part to keep in the string.
 */
@OptIn(ExperimentalStdlibApi::class)
public fun String.removeRange(startIndex: Int, endIndex: Int, replacement: String): String =
    (this as CharSequence).removeRange(range).toString()
```
(startsWith(prefix)) \{ n return substring(prefix.length)/n }\n return this\n\n/**\n * If this char sequence ends with the given [suffix], returns a new char sequence\n * with the suffix removed. Otherwise, returns a new char sequence with the same\n * characters.\n */\n\nfunc CharSequence.removeSuffix(suffix: CharSequence):\nCharSequence \{\n if (endsWith(suffix)) \{ n return subSequence(0, length - suffix.length)/n \} n return subSequence(0, length)/n\n\n/**\n * When this char sequence starts with the given [prefix] and ends with the given [suffix],\n * returns a new char sequence having both the given [prefix] and [suffix] removed.\n */\n\nfunc String.removeSuffix(suffix: CharSequence):\nString \{\n if (endsWith(suffix)) \{ n return substring(0, length - suffix.length)/n \}\n return this\n\n/**\n * If this string ends with the given [suffix], returns a new string unchanged.\n */\n\nfunc String.removeSuffix(suffix: CharSequence):\nString \{\n if (endsWith(suffix)) \{ n return this\n\n/**\n * Removes from a string both the [prefix] and [suffix] if and only if it starts with the [prefix] and ends\n * with the [suffix].\n */\n\nfunc String.removeSurrounding(prefix: CharSequence, suffix: CharSequence):\nString \{\n if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) \{\n n return subSequence(prefix.length, length - suffix.length)/n \}\n return subSequence(0, length)/n\n\n/**\n * When this char sequence starts with and ends with the given [delimiter],\n * returns a new char sequence having this [delimiter] removed both from the start and end.\n */\n\nfunc CharSequence.removeSurrounding(delimiter: CharSequence):\nCharSequence = removeSurrounding(delimiter, delimiter)/n\n\n/**\n * Removes the given [delimiter] string from both the start and the end of this string\n * if and only if it starts with and ends with the [delimiter].\n */\n\nfunc String.removeSurrounding(delimiter: CharSequence):\nString = removeSurrounding(delimiter, delimiter)/n\n\n/**\n * Replace part of string before the first occurrence of given delimiter with the [replacement] string.\n */\n\nfunc String.replaceBefore(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String \{\n val index = indexOf(delimiter)\n return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)/n\n\n/**\n * Replace part of string after the first occurrence of given delimiter with the [replacement] string.\n */\n\nfunc String.replaceAfter(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String \{\n val index = lastIndexOf(delimiter)\n return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)/n\n\n/**\n * Replace part of string after the last occurrence of given delimiter with the [replacement] string.\n */\n\nfunc String.replaceAfterLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String \{\n val index = lastIndexOf(delimiter)\n return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)/n\n\nOpen Source Used In DNAC Platform Arica  7989
return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)

return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)

/**
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 * Replace part of string before the last occurrence of given delimiter with the [replacement] string.
 */
public fun String.replaceBeforeLast(delimeter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimeter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Replace part of string before the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 * Replace part of string before the last occurrence of given delimiter with the [replacement] string.
 */
public fun String.replaceBeforeLast(delimeter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimeter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

// public fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean): String // JVM- and JS-specific
// public fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean): String // JVM- and JS-specific

/**
 * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression
 * with the given [replacement].
 * The [replacement] can consist of any combination of literal text and $-substitutions. To treat the replacement string
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.replace(regex: Regex, replacement: String): String = regex.replace(this, replacement)

/**
 * Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression
 * with the result of the given function [transform] that takes [MatchResult] and returns a string to be used as a
 * replacement for that match.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.replace(regex: Regex, noinline transform: (MatchResult) -> CharSequence): String =
    regex.replace(this, transform)

/**
 * Replaces the first occurrence of the given regular expression [regex] in this char sequence with specified [replacement] expression.
 * @param replacement A replacement expression that can include substitutions. See [Regex.replaceFirst] for details.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.replaceFirst(regex: Regex, replacement: String): String = regex.replaceFirst(this, replacement)

/**
 * Returns `true` if this char sequence matches the given regular expression.
 */
@kotlin.internal.InlineOnly
public inline infix fun CharSequence.matches(regex: Regex): Boolean = regex.matches(this)

/**
 * Implementation of [regionMatches] for CharSequences.
 * Invoked when it's already known that arguments are not Strings, so that no additional type checks are performed.
 */
internal fun CharSequence.regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean): Boolean {
    if ((otherOffset < 0) || (thisOffset < 0) || (thisOffset > this.length - length) ||
        (otherOffset > other.length - length)) {
        return false
    }

    for (index in 0 until length) {
        if
            (!this[thisOffset + index].equals(other[otherOffset + index], ignoreCase))
                return false
    }
    return true
}

/**
 * Returns `true` if this char sequence starts with the specified character.
 */
public fun CharSequence.startsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this.length > 0 &&
    this[0].equals(char, ignoreCase)

/**
 * Returns `true` if this char sequence ends with the specified character.
 */
public fun CharSequence.endsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this.length > 0 &&
    this[lastIndex].equals(char, ignoreCase)

/**
 * Returns `true` if a substring of this char sequence starting at the specified offset [startIndex] starts with the specified prefix.
 * @param prefix A prefix to search for.
 */
public fun CharSequence.startsWith(prefix: CharSequence, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if ((ignoreCase && this is String && prefix is String) ||
        (this.length > startIndex &&
        this.length - startIndex > prefix.length) &&
        this.regionMatchesImpl(startIndex, prefix, 0, prefix.length, ignoreCase))
    }

    return false
}

/**
 * Returns `true` if a substring of this char sequence starting at the specified offset [startIndex] starts with the specified prefix.
 * @param prefix A prefix to search for.
 */
public fun CharSequence.startsWith(prefix: String, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if ((ignoreCase && this is String && prefix is String) ||
        (this.length > startIndex &&
        this.length - startIndex > prefix.length) &&
        this.regionMatchesImpl(startIndex, prefix, 0, prefix.length, ignoreCase))
    }

    return false
}

/**
 * Returns the longest string `prefix` such that this char sequence starts with the specified prefix.
 * @param prefix A prefix to search for.
 */
public fun CharSequence.longestPrefixOf(prefix: CharSequence): Int =
    when (this)
        is String {
            this.indexOf(prefix, 0)
        } else {
            this.regionMatchesImpl(0, prefix, 0, prefix.length, ignoreCase)
        }

/**
 * Returns `true` if this char sequence ends with the specified suffix.
 * @param suffix A suffix to search for.
 */
public fun CharSequence.endsWith(suffix: CharSequence): Boolean {
    if ((ignoreCase && this is String && suffix is String) ||
        (this.length > 0 &&
        this.lastIndex - this.length + 1 > suffix.length) &&
        this.regionMatchesImpl(this.length - suffix.length, suffix, 0, suffix.length, ignoreCase))
    }

    return false
}

/**
 * Returns `true` if this char sequence contains the specified character.
 * @param char A character to search for.
 */
public fun CharSequence.containsChar(char: Char, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && char is String)
starting from the specified [startIndex] and optionally ignoring the case.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return a pair of an index of the first occurrence of matched string from [strings] and the string matched if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return a pair of an index of the last occurrence of matched string from [strings] and the string matched if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of matched string from [strings] or `-1` if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched string from [strings] or `-1` if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched character in [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of [string] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched string from [strings] or `-1` if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched character in [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of [string] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched string from [strings] or `-1` if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the first occurrence of [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

* by default `false`. To return an index of the last occurrence of matched character in [char] or `-1` if none is found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from
```kotlin
/*
 * Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 *
 * @param delimiters One or more characters to be used as delimiters.
 * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.
 * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.
 * @return An index of the last occurrence of the specified [string], starting from the specified [startIndex].
 */
@Suppress("INAPPLICABLE_OPERATOR_MODIFIER")
public operator fun CharSequence.contains(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Boolean =
    indexOf(char, startIndex, ignoreCase = ignoreCase) >= 0

/**
 * Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [string].
 *
 * @param ignoreCase `true` to ignore character case when matching a string. By default `false`.
 * @param startIndex The index of character to start searching at. The search proceeds backward toward the beginning of the string.
 * @return An index of the last occurrence of [string] or -1 if none is found.
 */
@Suppress("INAPPLICABLE_OPERATOR_MODIFIER")
public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =
    regex.containsMatchIn(this)

/**
 * Returns `true` if this char sequence contains the specified [string].
 *
 * @param ignoreCase `true` to ignore character case when comparing strings. By default `false`.
 * @return A boolean indicating whether the specified string or its case-insensitive version exists in the current sequence.
 */
@Suppress("INAPPLICABLE_OPERATOR_MODIFIER")
public operator fun CharSequence.contains(string: String, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Boolean =
    if (other is String)
        indexOfAny(other, startIndex, ignoreCase = ignoreCase) >= 0
    else
        nativeLastIndexOf(string, startIndex)

```
occurrences of the specified [delimiters].

* @param delimiters One or more strings to be used as delimiters.

* @param startIndex The index to start searching delimiters from. No range having its start value less than [startIndex] is returned. [startIndex] is coerced to be non-negative and not greater than length of this string.

* @param ignoreCase 'true' to ignore character case when matching a delimiter. By default 'false'.

* @param limit The maximum number of substrings to return. Zero by default means no limit is set. To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from the beginning to the end of this string, and finds at each position the first element in [delimiters] that matches this string at that position.

* @privated fun CharSequence.rangesDelimitedBy(delimiters: Array<out String>, startIndex: Int = 0, ignoreCase: Boolean = false, Int = 0): Sequence<IntRange> { require(limit >= 0, { "Limit must be non-negative, but was $limit." })

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

* @val delimitersList = delimiters.asList().return

Splits this char sequence to a sequence of strings around occurrences of the specified [delimiters].

* @param delimiters One or more strings to be used as delimiters.

* @param ignoreCase 'true' to ignore character case when matching a delimiter. By default 'false'.

* @param limit The maximum number of substrings to return. Zero by default means no limit is set. To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from the beginning to the end of this string, and matches at each position the first element in [delimiters] that is equal to a delimiter in this instance at that position.

* @privapublic fun CharSequence.split(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =

rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

Splits this char sequence to a list of strings around occurrences of the specified [delimiters].

* @param delimiters One or more strings to be used as delimiters.

* @param ignoreCase 'true' to ignore character case when matching a delimiter. By default 'false'.

* @param limit The maximum number of substrings to return. Zero by default means no limit is set. To avoid ambiguous results when strings in [delimiters] have characters in common, this method proceeds from the beginning to the end of this string, and matches at each position the first element in [delimiters] that is equal to a delimiter in this instance at that position.
listOf(this.toString())
    @
    val isLimited = limit > 0
    val result = ArrayList<String>(if (isLimited)
        limit.coerceAtMost(10) else 10)
    do {
        result.add(substring(currentOffset, nextIndex))
        currentOffset = nextIndex + delimiter.length
        // Do not search for next occurrence if we're reaching limit
        if (isLimited && result.size == limit - 1) break
        nextIndex = indexOf(delimiter, currentOffset, ignoreCase)
    } while (nextIndex != -1)
    result.add(substring(currentOffset, length))
    return result

/**
 * Splits this char
sequence around matches of the given regular expression.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.split(regex: Regex, limit: Int = 0): List<String> =
    regex.split(this, limit)

/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences: CRLF, LF or CR.
 */
public fun CharSequence.lineSequence(): Sequence<String> = splitToSequence("\r\n", "\n", "\r")

/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences: CRLF, LF or CR.
 */
public fun CharSequence.lines(): List<String> = lineSequence().toList()
internal constructor(@PublishedApi internal val storage: IntArray) : Collection<UInt> {

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(IntArray(size))

    /** Returns the array element at the given [index]. This method can be called using the
        index operator. */
    *fn * If the [index] is out of bounds of this array, throws an
        [IndexOutOfBoundsException] except in Kotlin/JS

    /** Sets the element at the given [index] to the specified value. */
    *fn fun get(index: Int): UInt = storage[index].toUInt() /**
        Returns the number of elements in the array. */
    *fn public override val size: Int get() = index.size

    /** Creates an iterator over the elements of the array. */
    *fn public override fun iterator(): UIntIterator = Iterator(storage)

    private class Iterator(private val array: IntArray) : UIntIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUInt() = if (index < array.size)
            array[index++].toUInt() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UInt): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is UInt) return false
        return storage.contains(element.toInt())
    }

    override fun containsAll(elements: Collection<UInt>): Boolean {
        return (elements as Collection<*>).all { it is UInt && storage.contains(it.toInt()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

//@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline class ULongArray
@PublishedApi
internal constructor(@PublishedApi internal val storage: ULongArray) : Collection<ULong> {

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(ULongArray(size))

    /** Returns the array element at the given [index]. */
    *fn operator fun get(index: Int): ULong = storage[index].toUInt() /**
        Returns the number of elements in the array. */
    *fn public override val size: Int get() = index.size

    /** Creates an iterator over the elements of the array. */
    *fn public override fun iterator(): ULongIterator = Iterator(storage)

    private class Iterator(private val array: ULongArray) : ULongIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextULong() = if (index < array.size)
            array[index++].toULong() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: ULong): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is ULong) return false
        return storage.contains(element.toLong())
    }

    override fun containsAll(elements: Collection<ULong>): Boolean {
        return (elements as Collection<*>).all { it is ULong && storage.contains(it.toDouble()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}
Open Source Used In DNAC Platform Arica  7999

```kotlin
Collection<ULong>): Boolean {null return (elements as Collection<?>).all { it is ULong &&
storage.contains(it.toLong()) }null override fun isEmpty(): Boolean = this.storage.size == 0null

Creates a new array of the specified [size], where each element is calculated by calling the specified
function. *null * The function [init] is called for each array element sequentially starting from the first one.
null It should return the value for an array element given its index.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {null return ULongArray(LongArray(size) { index ->
init(index).toLong()} }null

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
/** Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. null * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file. null */
Auto-generated file. DO NOT EDIT!\npackage kotlin\n@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline class UShortArray
@PublishedApi
@internal constructor(@PublishedApi internal val storage: ShortArray) : Collection<UShort> {null

Creates a new array of the specified [size], with all elements initialized to zero. null public constructor(size: Int): this(ShortArray(size))null /**null * Returns the array element at the given [index]. This method can be called
using the index operator. *null * If the [index] is out of bounds of this array, throws an
[IndexOutOfBoundsException] except in Kotlin/JS null * where the behavior is unspecified. null

public constructor(size: Int) : this(ShortArray(size))null /**null * Sets the element at the given
[index] to the given [value]. This method can be called using the index operator. *null * If the [index] is out of
bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS null * where the behavior is
unspecified. null

public constructor(size: Int) : this(ShortArray(size))null /**null * Returns the number of elements in the array. *null * If the [index] is out of
bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS null * where the behavior is
unspecified. null

public constructor(size: Int) : this(ShortArray(size))null /**null * Creates an iterator over the elements of the array. *null * public override operator fun iterator(): UShortIterator =
}null

private class Iterator(private val array: ShortArray) : UShortIterator() {

index = 0

override fun hasNext() = index < array.size

override fun nextUShort() = if (index < array.size) array[index++].toUShort() else throw NoSuchElementException(index.toString())null

override fun contains(element: UShort): Boolean {

// TODO: Eliminate this check after KT-30016 gets fixed. null // Currently JS BE does not generate special bridge method for this method.

if ((element as Any?) !is UShort)

return false

return storage.contains(element.toShort())null

override fun containsAll(elements: Collection<UShort>): Boolean {null return (elements as Collection<?>).all { it is UShort &&
storage.contains(it.toShort()) }null override fun isEmpty(): Boolean = this.storage.size == 0null

Creates a new array of the specified [size], where each element is calculated by calling the specified
function. *null * The function [init] is called for each array element sequentially starting from the first one.
null It should return the value for an array element given its index.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {null return UShortArray(ShortArray(size) { index ->
init(index).toShort()} }

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
/** Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. null * Use of this source code is governed by the Apache 2.0 license
that can be found in the license/LICENSE.txt file. null */
Auto-generated file. DO NOT EDIT!\n```
size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntArray.component1(): UInt {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun ULongArray.component1(): ULong {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UByteArray.component1(): UByte {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UShortArray.component1(): UShort {
    return get(0)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntArray.component2(): UInt {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun ULongArray.component2(): ULong {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UByteArray.component2(): UByte {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UShortArray.component2(): UShort {
    return get(1)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntArray.component3(): UInt {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun ULongArray.component3(): ULong {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UByteArray.component3(): UByte {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UShortArray.component3(): UShort {
    return get(2)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntArray.component4(): UInt {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun ULongArray.component4(): ULong {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UByteArray.component4(): UByte {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UShortArray.component4(): UShort {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UIntArray.component5(): UInt {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun ULongArray.component5(): ULong {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UByteArray.component5(): UByte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline operator fun UShortArray.component5(): UShort {
    return get(4)
}
behavior is unspecified.

```kotlin
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component4(): UInt \{ return get(3)\} throws 4th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component4(): ULong \{ return get(3)\} throws 4th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component4(): UByte \{ return get(3)\} throws 4th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component4(): UShort \{ return get(3)\} throws 4th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component5(): UInt \{ return get(4)\} throws 5th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component5(): ULong \{ return get(4)\} throws 5th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component5(): UByte \{ return get(4)\} throws 5th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component5(): UShort \{ return get(4)\} throws 5th *element* from the array.\n
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UIntArray.elementAt(index: Int): UInt

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun ULongArray.elementAt(index: Int): ULong

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UByteArray.elementAt(index: Int): UByte

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UShortArray.elementAt(index: Int): UShort
```

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UIntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UInt): UInt

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic expect fun UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort
```
samples.collections.Collections.Elements.elementAtOrElse
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {\n    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)\n}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun UIntArray.find(predicate: (UInt) -> Boolean): UInt? {
    return firstOrNull(predicate)\n}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun ULongArray.find(predicate: (ULong) -> Boolean): ULong? {
    return firstOrNull(predicate)\n}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun UByteArray.find(predicate: (UByte) -> Boolean): UByte? {
    return firstOrNull(predicate)\n}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.find
 * 
 * @sinceKotlin("1.3")
 * @experimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 */
public inline fun UShortArray.find(predicate: (UShort) -> Boolean): UShort? {
    return firstOrNull(predicate)\n}
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly

fun UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * @sample samples.collections.Collections.Elements.find
 */

fun ULongArray.findLast(predicate: (ULong) -> Boolean): ULong? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * @sample samples.collections.Collections.Elements.find
 */

fun UByteArray.findLast(predicate: (UByte) -> Boolean): UByte? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * @sample samples.collections.Collections.Elements.find
 */

fun UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? {
    return lastOrNull(predicate)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

fun UIntArray.first(): UInt {
    return storage.first().toUInt()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

fun ULongArray.first(): ULong {
    return storage.first().toULong()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

fun UByteArray.first(): UByte {
    return storage.first().toUByte()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

fun UShortArray.first(): UShort {
    return storage.first().toUShort()
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

fun UIntArray.first(predicate: (UInt) -> Boolean): UInt {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

fun ULongArray.first(predicate: (ULong) -> Boolean): ULong {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

fun UByteArray.first(predicate: (UByte) -> Boolean): UByte {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

fun UShortArray.first(predicate: (UShort) -> Boolean): UShort {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element, or `null` if the array is empty.
 */

fun UIntArray.firstOrNull(): UInt? {
    return if (isEmpty()) null else this[0]
}

/**
 * Returns the first element, or `null` if the array is empty.
 */
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULongArray.firstOrNull(): ULong? { if (isEmpty()) null else this[0] @ **n * Returns the first element, or `null` if the array is empty.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UByteArray.firstOrNull(): UByte? { if (isIsEmpty()) null else this[0] @ **n * Returns the first element, or `null` if the array is empty.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UShortArray.firstOrNull(): UShort? { if (isIsEmpty()) null else this[0] @ **n * Returns the first element matching the given [predicate], or `null` if element was not found.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UIntArray.firstOrNull(predicate: (UInt) -> Boolean): UInt? { if (isEmpty()) null else for (element in this) if (predicate(element)) return element @ **n * Returns the first element matching the given [predicate], or `null` if element was not found.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? { if (isEmpty()) null else for (element in this) if (predicate(element)) return element @ **n * Returns the first element matching the given [predicate], or `null` if element was not found.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? { if (isEmpty()) null else for (element in this) if (predicate(element)) return element @ **n * Returns the first element matching the given [predicate], or `null` if element was not found.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? { if (isEmpty()) null else for (element in this) if (predicate(element)) return element @ **n * Returns the first element matching the given [predicate], or `null` if element was not found.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt { return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) @ **n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt { return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) @ **n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public fun ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong { return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) @ **n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte { return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) @ **n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@@kotlin.internal.InlineOnly@public inline fun UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort { return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) @ **n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ ExperimentalUnsignedTypes@ public fun UIntArray.getOrElse(index: Int): UInt? { return if (index >= 0 && index <= lastIndex) get(index) else null @ **n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ ExperimentalUnsignedTypes@ public fun ULongArray.getOrElse(index: Int): ULong? { return if (index >= 0 && index <= lastIndex) get(index) else null @ **n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ ExperimentalUnsignedTypes@ public fun UByteArray.getOrElse(index: Int): UByte? { return if (index >= 0 && index <= lastIndex) get(index) else null @ **n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
*An@SinceKotlin("1.3")@ ExperimentalUnsignedTypes@ public fun UShortArray.getOrElse(index: Int): UShort? { return if (index >= 0 && index <= lastIndex) get(index) else null @ **n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
given [index] or `null` if the [index] is out of bounds of this array.

```kotlin
*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public fun UShortArray.getOrOrNull(index: Int):
UShort? {\n    return if (index >= 0 && index <= lastIndex) get(index) else null
}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UIntArray.indexOf(element: UInt): Int {\n    return storage.indexOfOf(element.toUInt()) \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun ULongArray.indexOf(element: ULong): Int {\n    return storage.indexOfOf(element.toULong()) \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UByteArray.indexOf(element: UByte): Int {\n    return storage.indexOfOf(element.toUByte()) \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UShortArray.indexOf(element: UShort): Int {\n    return storage.indexOfOf(element.toUShort()) \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toUInt()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toULong()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toUByte()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\n    return storage.indexOfFirst { predicate(it.toUShort()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UIntArray.lastIndexOf(predicate: (UInt) -> Boolean): Int {\n    return storage.lastIndexOf { predicate(it.toUInt()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun ULongArray.lastIndexOf(predicate: (ULong) -> Boolean): Int {\n    return storage.lastIndexOf { predicate(it.toULong()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UByteArray.lastIndexOf(predicate: (UByte) -> Boolean): Int {\n    return storage.lastIndexOf { predicate(it.toUByte()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun UShortArray.lastIndexOf(predicate: (UShort) -> Boolean): Int {\n    return storage.lastIndexOf { predicate(it.toUShort()) } \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public fun UIntArray.last(): UInt {\n    return storage.last().toUInt() \n}

*An@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public fun ULongArray.last(): ULong {\n    return storage.last().toULong() \n}
```

---

Open Source Used In DNAC Platform Arica 8005
@throws [NoSuchElementException] if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun ULongArray.last(): ULong {  
    return storage.last().toULong()  
}  
@throws [NoSuchElementException] if the array is empty.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UByteArray.last(): UByte {  
    return storage.last().toUByte()  
}  
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UShortArray.last(): UShort {  
    return storage.last().toUShort()  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UIntArray.last(predicate: (UInt) -> Boolean): UInt {  
    for (index in this.indices.reversed()) {  
        val element = this[index]  
        if (predicate(element)) return element  
    }  
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun ULongArray.last(predicate: (ULong) -> Boolean): ULong {  
    for (index in this.indices.reversed()) {  
        val element = this[index]  
        if (predicate(element)) return element  
    }  
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UByteArray.last(predicate: (UByte) -> Boolean): UByte {  
    for (index in this.indices.reversed()) {  
        val element = this[index]  
        if (predicate(element)) return element  
    }  
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UShortArray.last(predicate: (UShort) -> Boolean): UShort {  
    for (index in this.indices.reversed()) {  
        val element = this[index]  
        if (predicate(element)) return element  
    }  
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UIntArray.lastIndexOf(element: UInt): Int {  
    return storage.lastIndexOf(element.toInt())  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun ULongArray.lastIndexOf(element: ULong): Int {  
    return storage.lastIndexOf(element.toLong())  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UByteArray.lastIndexOf(element: UByte): Int {  
    return storage.lastIndexOf(element.toByte())  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ kotlinx.internal.InlineOnly
public inline fun UShortArray.lastIndexOf(element: UShort): Int {  
    return storage.lastIndexOf(element.toShort())  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.lastOrNull(): UInt? {  
    return if (isEmpty()) null else this[size - 1]  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.lastOrNull(): ULong? {  
    return if (isEmpty()) null else this[size - 1]  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.lastOrNull(): UByte? {  
    return if (isEmpty()) null else this[size - 1]  
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {  
    return if (isEmpty()) null else this[size - 1]  
}
```

```
Open Source Used In DNAC Platform Arica 8006
```
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or
 * `null` if no such element was found.
 */
@ExperimentalUnsignedTypes
public inline fun UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@ExperimentalUnsignedTypes
public inline fun ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@ExperimentalUnsignedTypes
public inline fun UByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@ExperimentalUnsignedTypes
public inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@ExperimentalUnsignedTypes
public fun UIntArray.random(random: Random): UInt {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@ExperimentalUnsignedTypes
public fun ULongArray.random(random: Random): ULong {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.random(random: Random): UByte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.random(random: Random): UShort {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}
randomOrNull(Random): ULong? * Returns a random element from this array, or `null` if this array is empty.

@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(): UByte? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(): UShort? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(random: Random): UByte? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(random: Random): UShort? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(): UInt {
    return storage.single().toUInt()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(): ULong {
    return storage.single().toULong()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.single(): UByte {
    return storage.single().toUByte()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.single(): UShort {
    return storage.single().toUShort()
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    return single
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(predicate: (ULong) -> Boolean): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    return single
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    return single
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3") @ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    return single
}
for (element in this) {
    if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
    }
}
if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

@Suppress("UNCHECKED_CAST")
    return single as ULong
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    @Suppress("UNCHECKED_CAST")
        return single as UByte
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

    @Suppress("UNCHECKED_CAST")
    return single as UShort
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public fun UIntArray.singleOrNull(): UInt? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public fun ULongArray.singleOrNull(): ULong? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public fun UByteArray.singleOrNull(): UByte? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 * @since kotlin.
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @public fun UShortArray.singleOrNull(): UShort? {
    return if (size == 1) this[0] else null
}

/**
 * Returns a list
containing all elements except first \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\public fun UIntArray.drop(n: Int): List\<UInt> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return takeLast((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except first \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\public fun ULongArray.drop(n: Int): List\<ULong> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return takeLast((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except first \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\public fun UByteArray.drop(n: Int): List\<UByte> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return takeLast((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except first \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\public fun UShortArray.drop(n: Int): List\<UShort> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return takeLast((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except first \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UIntArray.dropLast(n: Int): List\<UInt> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return take((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun ULongArray.dropLast(n: Int): List\<ULong> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return take((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UByteArray.dropLast(n: Int): List\<UByte> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return take((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UShortArray.dropLast(n: Int): List\<UShort> {\n    require(n >= 0) { \"Requested element count \([n]\) is less than zero.\" }\n    return take((size - n).coerceAtLeast(0))\n}***\n* \n* \n* Returns a list containing all elements except last \([n]\) elements.\n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UIntArray.dropLastWhile(predicate: (UInt) -> Boolean): List\<UInt> {\n    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}***\n* \n* \n* Returns a list containing all elements except last elements that satisfy the given predicate.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun ULongArray.dropLastWhile(predicate: (ULong) -> Boolean): List\<ULong> {\n    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}***\n* \n* \n* Returns a list containing all elements except last elements that satisfy the given predicate.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List\<UByte> {\n    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}***\n* \n* \n* Returns a list containing all elements except last elements that satisfy the given predicate.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

*\@SinceKotlin("1.3\")\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\public inline fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List\<UShort> {\n    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}***\n* \n* \n* Returns a list containing all elements except last elements that satisfy the given predicate.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample \n * sample \n * samples \n * \n * @throws \n * IllegalArgumentException if \([n]\) is negative.\n * \n * @sample

Open Source Used In DNAC Platform Arica 8010
Returns a list containing all elements except last elements that satisfy the given [predicate].

- **UByteArray.dropLastWhile** (predicate: (UByte) -> Boolean): List<UByte>

- **UShortArray.dropLastWhile** (predicate: (UShort) -> Boolean): List<UShort>

- **UIntArray.dropWhile** (predicate: (UInt) -> Boolean): List<UInt>

- **ULongArray.dropWhile** (predicate: (ULong) -> Boolean): List<ULong>

- **UByteArray.dropWhile** (predicate: (UByte) -> Boolean): List<UByte>

- **UShortArray.dropWhile** (predicate: (UShort) -> Boolean): List<UShort>

- **UIntArray.filter** (predicate: (UInt) -> Boolean): List<UInt>

- **ULongArray.filter** (predicate: (ULong) -> Boolean): List<ULong>

- **UByteArray.filter** (predicate: (UByte) -> Boolean): List<UByte>

- **UShortArray.filter** (predicate: (UShort) -> Boolean): List<UShort>
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {
    return filterIndexedTo(ArrayList<UInt>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), predicate)
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}
samples.collections.Collections.Filtering.filterIndexedTo

*\n@SinceKotlin("1.3."")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <C : MutableCollection\<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element))
            destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {
    return filterNotTo(ArrayList<UInt>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {
    return filterNotTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {
    return filterNotTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @sample samples.collections.Collections.Filtering.filter
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {
    return filterNotTo(ArrayList<UShort>(), predicate)
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection\<in UInt>> UIntArray.filterNotTo(destination: C, predicate: (UInt) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection\<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection\<in UByte>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterTo
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection\<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}
mutable.

```kotlin
MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * 
 * @sample samples.collections.Collections.Filtering.filterTo
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * @kotlin.internal.InlineOnly
 */
public fun <C : MutableCollection<in UByte>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UIntArray.slice(indices: IntRange): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun ULongArray.slice(indices: IntRange): List<ULong> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UByteArray.slice(indices: IntRange): List<UByte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UShortArray.slice(indices: IntRange): List<UShort> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UIntArray.slice(indices: Iterable<Int>): List<UInt> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UInt>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun ULongArray.slice(indices: Iterable<Int>): List<ULong> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<ULong>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns an array containing elements of this array at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UIntArray.sliceArray(indices: Collection<Int>): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun ULongArray.sliceArray(indices: Collection<Int>): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UByteArray.sliceArray(indices: Collection<Int>): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 * 
 * @sinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @public fun UShortArray.sliceArray(indices: Collection<Int>): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}
```
Collection<Int>): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sliceArray(indices: IntRange): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: IntRange): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

/**
 * Returns a list containing first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.take(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UInt>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.takeLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UInt>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.takeLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.takeLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = this.size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UShort>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.takeLastWhile(predicate: (ULong) -> Boolean): List<ULong> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.takeWhile(predicate: (UInt) -> Boolean): List<UInt> {
    val list = ArrayList<UInt>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.takeWhile(predicate: (ULong) -> Boolean): List<ULong> {
    val list = ArrayList<ULong>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.reversed(): List<UShort> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reversedArray(): UIntArray {
    return UIntArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reversedArray(): ULongArray {
    return ULongArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reversedArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}

/**
 * Returns an array with elements of this array in reversed order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reversedArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}
according to their natural sort order.

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sorted(): List<UInt> {
    return copyOf().apply { sort() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sorted(): List<ULong> {
    return copyOf().apply { sort() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.sorted(): List<UByte> {
    return copyOf().apply { sort() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sorted(): List<UShort> {
    return copyOf().apply { sort() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortedArrayDescending(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sortedArray(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }.toMutableList()
}

/**
 * Returns a list of all elements sorted descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortedArrayDescending(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sortedArrayDescending(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }.asList().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sortedDescending(): List<UInt> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sortedDescending(): List<ULong> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.sortedDescending(): List<UByte> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sortedDescending(): List<UShort> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sortedArrayDescending(): List<UInt> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sortedArrayDescending(): List<ULong> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.sortedArrayDescending(): List<UByte> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sortedArrayDescending(): List<UShort> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sortedDescending(): List<UInt> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sortedDescending(): List<ULong> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.sortedDescending(): List<UByte> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sortedDescending(): List<UShort> {
    return copyOf().apply { sortDescending() }.reversed().toMutableList() * SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}

/**
 * Returns a list of all elements sorted ascending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}

/**
 * Returns a list of all elements sorted ascending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortedArray(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}

/**
 * Returns a list of all elements sorted ascending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortedArray(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}

/**
 * Returns a list of all elements sorted ascending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortedArray(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }.toMutableList()
}
List<?uByte> {
    return copyOf().apply { sort() }.reversed()
} * Returns a list of all elements sorted descending according to their natural sort order. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public fun UShortArray.sortedDescending(): List<UShort> {
    return copyOf().apply { sort() }.reversed()
} * Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UByteArray.asList(): List<UByte> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asList(): List<UShort> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun ULongArray.asList(): List<ULong> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asByteArray(): ByteArray * Returns an array of type [ByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asByteArray(): UByteArray * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asIntArray(): IntArray * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asIntArray(): UShortArray * Returns an array of type [UShortArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asList(): List<UShort> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun ULongArray.asList(): List<ULong> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asByteArray(): ByteArray * Returns an array of type [ByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asByteArray(): UByteArray * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asIntArray(): IntArray * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asIntArray(): UShortArray * Returns an array of type [UShortArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asList(): List<UShort> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun ULongArray.asList(): List<ULong> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asByteArray(): ByteArray * Returns an array of type [ByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asByteArray(): UByteArray * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asIntArray(): IntArray * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asIntArray(): UShortArray * Returns an array of type [UShortArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UShortArray.asList(): List<UShort> * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
@public expect fun UIntArray.asList(): List<UInt> * Returns a [List] that wraps the original array.
fun ULongArray.contentEquals(other: ULongArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun UByteArray.contentEquals(other: UByteArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun UShortArray.contentEquals(other: UShortArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun UIntArray?.contentEquals(other: UIntArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun ULongArray?.contentEquals(other: ULongArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun UByteArray?.contentEquals(other: UByteArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
fun UShortArray?.contentEquals(other: UShortArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun UIntArray.contentHashCode(): Int {
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun ULongArray.contentHashCode(): Int {
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun UByteArray.contentHashCode(): Int {
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun UShortArray.contentHashCode(): Int {
    return this.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun UIntArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun ULongArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is `[List]`.
 */
fun UByteArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()
}
return this?.storage.contentHashCode() // Returns a hash code based on the contents of this array as if it is [List].

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray?.contentToString(): String {
    return this?.contentToString() ?: "null"
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.contentToString(): String {
    return this?.contentToString() ?: "null"
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.contentToString(): String {
    return this?.contentToString() ?: "null"
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.contentToString(): String {
    return this?.contentToString() ?: "null"
}

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.contentToString(): String {
    return this?.contentToString() ?: "null"
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when [startIndex] > [endIndex].
 * @param destinationOffset the position in the [destination] array to copy to.
 * @param destination the array to copy to. 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @param destination the array to copy to. 0 by default.
 * @param destinationOffset the position in the [destination] array to copy to.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalStateException] when [startIndex] or [endIndex] is out of range of this array indices or when [startIndex] > [endIndex].
 * @param destination the array to copy to. 0 by default.
 * @param destinationOffset the position in the [destination] array to copy to.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @param destination the array to copy to. 0 by default.
 * @param destinationOffset the position in the [destination] array to copy to.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalStateException] when [startIndex] or [endIndex] is out of range of this array indices or when [startIndex] > [endIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun
UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UIntArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return destination\n}/*\n* Copies this array or its subrange into the [destination] array and returns that array.
* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* @return the [destination] array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UIntArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return destination\n}/*\n* Copies this array or its subrange into the [destination] array and returns that array.\n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* @return the [destination] array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
ULongArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return destination\n}/*\n* Copies this array or its subrange into the [destination] array and returns that array.\n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* @return the [destination] array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UByteArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return destination\n}/*\n* Copies this array or its subrange into the [destination] array and returns that array.\n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* @return the [destination] array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UShortArray {\n    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)\n    return destination\n}/*\n* Copies this array or its subrange into the [destination] array and returns that array.\n* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n* @param destination the array to copy to.\n* @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.\n* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.\n* @return the [destination] array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UByteArray.copyOf(): UByteArray {\n    return UByteArray(storage.copyOf())\n}/*\n* Returns new array which is a copy of the original array.\n* @sample samples.collections.Arrays.CopyOfOperations.copyOf\n* \n* @return new array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UIntArray.copyOf(): UIntArray {\n    return UIntArray(storage.copyOf())\n}/*\n* Returns new array which is a copy of the original array.\n* @sample samples.collections.Arrays.CopyOfOperations.copyOf\n* \n* @return new array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun ULongArray.copyOf(): ULongArray {\n    return ULongArray(storage.copyOf())\n}/*\n* Returns new array which is a copy of the original array.\n* @sample samples.collections.Arrays.CopyOfOperations.copyOf\n* \n* @return new array.*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\ninline fun UByteArray.copyOf(): UByteArray {\n    return UByteArray(storage.copyOf())\n}/*\n* Returns new array which is a copy of the original array.\n* @sample samples.collections.Arrays.CopyOfOperations.copyOf\n* \n* @return new array.*/
which is a copy of the original array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.copyOf(): UShortArray { return UShortArray(storage.copyOf()) } // Returns new array which is a copy of the original array, resized to the given [newSize].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.copyOf(newSize: Int): UIntArray { return UIntArray(storage.copyOf(newSize)) } // Returns new array which is a copy of the original array, resized to the given [newSize].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.copyOf(newSize: Int): ULongArray { return ULongArray(storage.copyOf(newSize)) } // Returns new array which is a copy of the original array, resized to the given [newSize].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.copyOf(newSize: Int): UByteArray { return UByteArray(storage.copyOf(newSize)) } // Returns new array which is a copy of the original array, resized to the given [newSize].
```

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
specified range of the original array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray {
    return UShortArray(storage.copyOfRange(fromIndex, toIndex))
}
```

```kotlin
/**
 * Fills this array or its subrange with the specified [element] value.
 * If [fromIndex] is less than zero or [toIndex] is greater than the size of this array,
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of the array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.fill(element: UInt, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toInt(), fromIndex, toIndex)
}
```

```kotlin
/**
 * Fills this array or its subrange with the specified [element] value.
 * If [fromIndex] is less than zero or [toIndex] is greater than the size of this array,
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of the array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toLong(), fromIndex, toIndex)
}
```

```kotlin
/**
 * Fills this array or its subrange with the specified [element] value.
 * If [fromIndex] is less than zero or [toIndex] is greater than the size of this array,
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of the array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toByte(), fromIndex, toIndex)
}
```

```kotlin
/**
 * Fills this array or its subrange with the specified [element] value.
 * If [fromIndex] is less than zero or [toIndex] is greater than the size of this array,
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of the array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toShort(), fromIndex, toIndex)
}
```

```kotlin
/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UIntArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun ULongArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UShortArray.indices: IntRange
    get() = storage.indices

/**
 * Returns an array containing all elements of the original array and then the given
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UIntArray.plus(element: UInt): UIntArray {
    return UIntArray(storage +
    element.toInt())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage +
    element.toLong())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage +
    element.toByte())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UShortArray.plus(element: UShort): UShortArray {
    return UShortArray(storage +
    element.toShort())
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [collections].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator
fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result =
    storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return
    UIntArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [collections].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator
fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result =
    storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return
    ULongArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [collections].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator
fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result =
    storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return
    UByteArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [collections].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator
fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result =
    storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return
    UShortArray(result)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UIntArray.plus(elements: Collection<UInt>): UIntArray {
    return UIntArray(storage +
    elements.storage)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
ULongArray.plus(elements: Collection<ULong>): ULongArray {
    return ULongArray(storage +
    elements.storage)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UByteArray.plus(elements: Collection<UByte>): UByteArray {
    return UByteArray(storage +
    elements.storage)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UShortArray.plus(elements: Collection<UShort>): UShortArray {
    return UShortArray(storage +
    elements.storage)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UIntArray.sort(): Unit {
    if (size > 1)
    sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
ULongArray.sort(): Unit {
    if (size > 1)
    sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UByteArray.sort(): Unit {
    if (size > 1)
    sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun
UShortArray.sort(): Unit {
    if (size > 1)
    sortArray(this, 0, size)
}
samples.collections.Arrays.Sorting.sortArray
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1) sortArray(this, 0, size)
}

/**
 * Sorts the array in-place.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1) sortArray(this, 0, size)
}

/**
 * Sorts a range in the array in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    sortArray(this, fromIndex, toIndex)
    sort(fromIndex, toIndex, sortArray)
}

/**
 * Sorts a range in the array in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
    sort(fromIndex, toIndex, sortArray)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortDescending
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex, sortArray)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * 
 * @sample samples.collections.Arrays.Sorting.sortDescending
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex, sortArray)
}
greater than the size of this array.\n
 stayed are sorted descending according to their natural sort order.\n
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.sortDescending(fromIndex: Int, toIndex: Int): Unit {\n    return storage.copyOf()\n
 * Returns an array of type [UByteArray], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.\n
 * @SinceKotlin("1.3")\n
 UByteArray.toByteArrayList(): ByteArrayList {\n    return storage.copyOf()\n
 * Returns a *typed* object array containing all of the elements of this array.

 * @SinceKotlin("1.3")\n
 UByteArray.toByteArrayList(): ByteArrayList {\n    return storage.copyOf()\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * @SinceKotlin("1.3")\n
 * Returns a *typed* object array containing all of the elements of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [UByteArray], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

 * @SinceKotlin("1.3")\n
 * Returns an array of type [ByteArrayList], which is a copy of this array where each element is a signed reinterpretation of the corresponding element of this array.
Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.

If any two elements are equal, the last one overwrites the former value in the map.

@sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in ULong, in V>> ULongArray.associateWithTo(destination: M, valueSelector: (ULong) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UByte, in V>> UByteArray.associateWithTo(destination: M, valueSelector: (UByte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array, where key is the element itself and value is provided by the [valueSelector] function applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 * @sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UShort, in V>> UShortArray.associateWithTo(destination: M, valueSelector: (UShort) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * and its index in the original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.flatMapIndexed(transform: (index: Int, UInt) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.flatMapIndexed(transform: (index: Int, ULong) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.flatMapIndexed(transform: (index: Int, UByte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMapIndexed

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.flatMapIndexed(transform: (index: Int, UShort) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array.\(n\) * \(n\) *

```
@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun \u003cR>
ULongArray.flatMapIndexed(transform: \(\text{fn} \to \text{List}\langle\text{R}\rangle\)): \text{List}\langle\text{R}\rangle |\(n\) * \(n\) * Returns a single list of all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array.\(n\) * \(n\) *

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R>
UByteArray.flatMapIndexed(transform: \(\text{fn} \to \text{List}\langle\text{R}\rangle\)): \text{List}\langle\text{R}\rangle |\(n\) * \(n\) * Returns a single list of all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array.\(n\) * \(n\) *

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R>
UShortArray.flatMapIndexed(transform: \(\text{fn} \to \text{List}\langle\text{R}\rangle\)): \text{List}\langle\text{R}\rangle |\(n\) * \(n\) * Returns a single list of all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array.\(n\) * \(n\) *

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R, C :
MutableCollection\(\in\text{R}\)> UIntArray.flatMapIndexedTo\(\text{destination}:\text{C},\text{transform}:\text{fn}\to\text{Iterable}\langle\text{R}\rangle\)): \text{C} |\(n\) var \text{index} = 0 |\(n\) for \text{element} in \text{this} |\(n\) val \text{list} = \text{transform}(\text{index}++, \text{element})|\(n\) \text{destination}.add\text{All}(\text{list}) |\(n\) * \(n\) * \text{Appends all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array, to the given \text{destination}\).\(n\)

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R, C :
MutableCollection\(\in\text{R}\)> ULongArray.flatMapIndexedTo\(\text{destination}:\text{C},\text{transform}:\text{fn}\to\text{Iterable}\langle\text{R}\rangle\)): \text{C} |\(n\) var \text{index} = 0 |\(n\) for \text{element} in \text{this} |\(n\) val \text{list} = \text{transform}(\text{index}++, \text{element})|\(n\) \text{destination}.add\text{All}(\text{list}) |\(n\) * \(n\) * \text{Appends all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array, to the given \text{destination}\).\(n\)

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R, C :
MutableCollection\(\in\text{R}\)> UByteArray.flatMapIndexedTo\(\text{destination}:\text{C},\text{transform}:\text{fn}\to\text{Iterable}\langle\text{R}\rangle\)): \text{C} |\(n\) var \text{index} = 0 |\(n\) for \text{element} in \text{this} |\(n\) val \text{list} = \text{transform}(\text{index}++, \text{element})|\(n\) \text{destination}.add\text{All}(\text{list}) |\(n\) * \(n\) * \text{Appends all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array, to the given \text{destination}\).\(n\)

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R, C :
MutableCollection\(\in\text{R}\)> UShortArray.flatMapIndexedTo\(\text{destination}:\text{C},\text{transform}:\text{fn}\to\text{Iterable}\langle\text{R}\rangle\)): \text{C} |\(n\) var \text{index} = 0 |\(n\) for \text{element} in \text{this} |\(n\) val \text{list} = \text{transform}(\text{index}++, \text{element})|\(n\) \text{destination}.add\text{All}(\text{list}) |\(n\) * \(n\) * \text{Appends all elements yielded from results of \[\text{transform}\] function being invoked on each element\(n\) and its index in the original array, to the given \text{destination}\).\(n\)

@sample samples.collections.Collections.Transformations.flatMapIndexed

/*\@SinceKotlin("1.3")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\u003cn@ExperimentalUnsignedTypes\u003cs@kotlin.internal.InlineOnly\u003epublic inline fun <R, C :
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
UIntArray.groupByTo(destination: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<UInt>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
UByteArray.groupByTo(destination: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<UByte>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
UShortArray.groupByTo(destination: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<UShort>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
ULongArray.groupByTo(destination: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<ULong>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
ByteArray.groupByTo(destination: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<Byte>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
ByteArray.groupByTo(destination: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<Byte>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
ByteArray.groupByTo(destination: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<Byte>()
        }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * 
 * @return The [destination] map.
 * 
 * @sample samples.collections.Collections.Transformations.groupBy
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
ByteArray.groupByTo(destination: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<Byte>()
        }
        list.add(element)
    }
    return destination
}
original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements. Returns the [destination] map.

*.samples.collections.Collections.Transformations.groupBy

<@ Since Kotlin("1.3")> @ExecutableUnsignedTypes@<@ in kotlin.internal.InlineOnly@<@> public inline fun <K, M : MutableMap<in K, MutableList<UShort>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K):

M {
    for (element in this)

        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<UShort>()
        }
        list.add(element)
    }
    return destination
}

// Group by key and values
@SinceKotlin("1.3")
@ExecutableUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K,
valueTransform: (UInt) -> V): M {

    for (element in this)

        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}

@SinceKotlin("1.3")
@ExecutableUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K,
valueTransform: (ULong) -> V): M {

    for (element in this)

        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}

@SinceKotlin("1.3")
@ExecutableUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K,
valueTransform: (UByte) -> V): M {

    for (element in this)

        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}

@SinceKotlin("1.3")
@ExecutableUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K,
valueTransform: (UShort) -> V): M {

    for (element in this)

        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}
element in the original array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R>
UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size),
        transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R>
UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size),
        transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R>
UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline
fun <R> ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline
fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline
fun <R> UShortArray.mapIndexed(transform: (index: Int, UShort) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C :
MutabelCollection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (index: Int, UInt) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C :
MutabelCollection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (index: Int, ULong) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C :
MutabelCollection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C :
MutabelCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```
element and the element itself * and returns the result of the transform applied to the element.

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (Int, UShort) -> R): C {
\n    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (ULong) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UByte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UIntArray.withIndex(): Iterable<IndexedValue<UInt>> {
    return IndexingIterable { iterator() }
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun ULongArray.withIndex(): Iterable<IndexedValue<ULong>> {
    return IndexingIterable { iterator() }
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UByteArray.withIndex(): Iterable<IndexedValue<UByte>> {
    return IndexingIterable { iterator() }
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UShortArray.withIndex(): Iterable<IndexedValue<UShort>> {
    return IndexingIterable { iterator() }
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UIntArray.all(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```
false
    return true
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if at least one element matches
 * the given [predicate].
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.any(predicate: (UChar) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches
 * the given [predicate].
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.any(predicate: (UChar) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements matching the given
 * [predicate].
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.count(predicate: (UChar) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given
 * [predicate].
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UCharArray.count(predicate: (UChar) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and
 * applying [operation] from left to right from each
 * element.
 * 
 * @param [operation] function that takes current accumulator
 * value and an element, and calculates the next accumulator
 * value and an element, and calculates the next accumulator
 * value.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
    UCharArray.fold(initial: R, operation: (acc: R, UChar) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying
 * [operation] from left to right from each
 * element.
 * 
 * @param [operation] function that takes current accumulator
 * value and an element, and calculates the next accumulator
 * value and an element, and calculates the next accumulator
 * value.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
    UCharArray.fold(initial: R, operation: (acc: R, UChar) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
public inline fun <R> ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
next accumulator value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldRightIndexed(initial: R, operation: (index: Int, UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldRightIndexed(initial: R, operation: (index: Int, ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * Returns the specified [initial] value if the array is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.foldRightIndexed(initial: R, operation: (index: Int, UShort, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}
```
value and applying [operation] from right to left.  For each element with its index in the original array and current accumulator value. Returns the specified [initial] value if the array is empty. @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.forEach(action: (UInt) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.forEach(action: (ULong) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.forEach(action: (UByte) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.forEach(action: (UShort) -> Unit): Unit {
    for (element in this) action(element)
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.max(): UInt? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.max(): ULong? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.max(): UByte? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.max(): UShort? {
    return maxOrNull()
}
```

Open Source Used In DNAC Platform Arica  8040
ReplaceWith("maxOrNull()")\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.max(): UShort? {\n    return maxOrNull()\n}\n
ReplaceWith("maxByOrNull(selector)")\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxBy(selector: (UByte) -> R): UByte? {\n    return maxByOrNull(selector)\n} \n
ReplaceWith("maxByOrNull(selector)")\n@DeprecatedSinceKotlin(warningSince = "1.4")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {\n    return maxByOrNull(selector)\n}

/**
 * Returns the first element yielding the largest value of the given function or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UIntArray.maxByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> ULongArray.maxByOrNull(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UByteArray.maxByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.maxByOrNull
 */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>> UShortArray.maxByOrNull(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function\n * applied to each element in the array.\n * @throws
 */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun |
NoSuchElementException if the array is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.maxOf(selector: (UInt) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.maxOf(selector: (ULong) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.maxOf(selector: (UByte) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.maxOf(selector: (UShort) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.maxOf(selector: (UInt) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.maxOf(selector: (ULong) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.maxOf(selector: (UByte) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.maxOf(selector: (UShort) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```
maxOf(maxValue, v) return maxValue

// Returns the largest value among all values produced by [selector] function\n* @throws NoSuchElementException if the array is empty.

*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun UShortArray.maxOf(selector: (UShort) -> Float): Float 

if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

// Returns the largest value among all values produced by [selector] function\n* applied to each element in the array.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun <R : Comparable<R>> UIntArray.maxOf(selector: (UInt) -> R): R 

if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

// Returns the largest value among all values produced by [selector] function\n* applied to each element in the array.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun <R : Comparable<R>> ULongArray.maxOf(selector: (ULong) -> R): R 

if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

// Returns the largest value among all values produced by [selector] function\n* applied to each element in the array or `null` if there are no elements.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun <R : Comparable<R>> UByteArray.maxOf(selector: (UByte) -> R): R 

if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

// Returns the largest value among all values produced by [selector] function\n* applied to each element in the array or `null` if there are no elements.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun <R : Comparable<R>> UShortArray.maxOf(selector: (UShort) -> R): R 

if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue

// Returns the largest value among all values produced by [selector] function\n* applied to each element in the array or `null` if there are no elements.\n
*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution

ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
public inline fun

Open Source Used In DNAC Platform Arica 8043
ULongArray.maxOfOrNull(selector: (ULong) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Float): Float? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Float): Float? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Float): Float? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Float): Float? {
    var maxValue = selector(this[0])
    for (i in 1 until lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
Comparable<R>> UIntArray.maxOfOrNull(selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxOfOrNull(selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxOfWith(comparator: Comparator<in R>, selector: (ULong) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values
 * produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic inline fun <R> UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    return maxOfWithOrNull(comparator, selector)
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced
 * by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic inline fun <R> UIntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    return maxOfWithOrNull(comparator, selector)
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic inline fun <R> ULongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    return maxOfWithOrNull(comparator, selector)
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic fun UIntArray.maxOrNull(): UInt? {
    if (isEmpty()) return null
    return maxOrNull()
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic fun ULongArray.maxOrNull(): ULong? {
    if (isEmpty()) return null
    return maxOrNull()
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic fun UByteArray.maxOrNull(): UByte? {
    if (isEmpty()) return null
    return maxOrNull()
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
By LambdaReturnType\n@ExperimentalUnsignedTypes\n@otlin.internal.InlineOnly\n\npublic fun UShortArray.maxOrNull(): UShort? {
    if (isEmpty()) return null
    return maxOrNull()
}
```kotlin
instead.
ReplaceWith("\n\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxWith(comparator: Comparator<in ULong>): ULong? {\n    return maxWithOrNull(comparator)\n}\n\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {\n    return maxWithOrNull(comparator)\n}\n\n\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {\n    return maxWithOrNull(comparator)\n}\n\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.maxWithOrNull(comparator: Comparator<in UInt>): UInt? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWithOrNull(comparator: Comparator<in UByte>): UByte? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWithOrNull(comparator: Comparator<in UShort>): UShort? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.maxWithOrNull(comparator: Comparator<in ULong>): ULong? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.maxWithOrNull(comparator: Comparator<in UByte>): UByte? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.maxWithOrNull(comparator: Comparator<in UShort>): UShort? {\n    if (isEmpty()) return null\n    var max = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(max, e) < 0) max = e
    }\n    return max
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.min(): ULong? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.min(): UByte? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.min(): UShort? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.min(): UInt? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.min(): ULong? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.min(): UByte? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.min(): UShort? {
    return minOrNull()
}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    return minByOrNull(selector)\n}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    return minByOrNull(selector)\n}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    return minByOrNull(selector)\n}\n\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    return minByOrNull(selector)\n}\n```
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> UByteArray.minBy(selector: (UByte) -> R): UByte? {\n    return minByOrNull(selector)\n} @Deprecated("Use minByOrNull instead.\")
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> UShortArray.minBy(selector: (UShort) -> R): UShort? {\n    return minByOrNull(selector)\n} @Deprecated("Use minByOrNull instead.\")
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} @Deprecated("Use minByOrNull instead.\")
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} @Deprecated("Use minByOrNull instead.\")
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} @Deprecated("Use minByOrNull instead.\")
ReplaceWith("minByOrNull(selector)\")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R : Comparable<R>> UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
} @Deprecated("Use minByOrNull instead.\")
var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UByteArray.minOf(selector: (UByte) -> Double): Double {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UShortArray.minOf(selector: (UShort) -> Double): Double {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UIntArray.minOf(selector: (UInt) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun ULongArray.minOf(selector: (ULong) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UByteArray.minOf(selector: (UByte) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UShortArray.minOf(selector: (UShort) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun UIntArray.minOf(selector: (UInt) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function\n * applied to each element in the array.\n * \n * @throws NoSuchElementException if the array is empty.\n * @since Kotlin("1.4")\n * @optIn(kotlin.experimental.ExperimentalTypeInference::class)\n * @overloadResolution ByLambdaReturnType\n * @experimentalUnsignedTypes\n * @kotlin.internal.InlineOnly\n * @public inline fun ULongArray.minOf(selector: (ULong) -> Float): Float {
 var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
Open Source Used In DNAC Platform Arica 8050
*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UShortArray.minOfOrNull(selector: (UShort) -> Double): Double? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UIntArray.minOfOrNull(selector: (UInt) -> Float): Float? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

ULongArray.minOfOrNull(selector: (ULong) -> Float): Float? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UByteArray.minOfOrNull(selector: (UByte) -> Float): Float? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UShortArray.minOfOrNull(selector: (UShort) -> Float): Float? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UIntArray.minOfOrNull(selector: (UInt) -> R): R? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

ULongArray.minOfOrNull(selector: (ULong) -> R): R? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UByteArray.minOfOrNull(selector: (UByte) -> R): R? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function\n * applied to each element in the array or `null` if there are no elements.
 * \n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */

*\@SinceKotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution

ByLambdaReturnType\n*\@ExperimentalUnsignedTypes\n*\@kotlin.internal.InlineOnly\n*\@public inline fun

UShortArray.minOfOrNull(selector: (UShort) -> R): R? { \n    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) { \n        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
Comparable<R>> UByteArray.minOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var
    minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or 'null' if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UIntArray.minOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UByteArray.minOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UShortArray.minOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
function applied to each element in the array or ‘null’ if there are no elements.

```kotlin
*\@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

public inline fun <R>
UShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {
    if (isEmpty())
        return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector]
function applied to each element in the array or ‘null’ if there are no elements.

*\@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 UBArray.minOfWithOrNull(comparator: Comparator<in UByte>, selector: (UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest element or ‘null’ if there are no elements.

*\@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 UIntArray.minOrNull(): UInt?

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 public fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 public fun ULongArray.minWith(comparator: Comparator<in ULong>): ULong? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 public fun UByteArray.minWith(comparator: Comparator<in UByte>): UByte? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)

 public fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {
    return minWithOrNull(comparator)
}
```

Open Source Used In DNAC Platform Arica 8053
Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\nfun UIntArray.minWithOrNull(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\nfun ULongArray.minWithOrNull(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\nfun UByteArray.minWithOrNull(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\nfun UShortArray.minWithOrNull(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

Since Kotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.none(): Boolean {
    return isEmpty()
}

Since Kotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.none(): Boolean {
    return isEmpty()
}

Since Kotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.none(): Boolean {
    return isEmpty()
}

Since Kotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.none(): Boolean {
    return isEmpty()
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.onEach(action: (UInt) -> Unit): UIntArray {
    return apply { for (element in this)
        action(element) }
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.onEach(action: (ULong) -> Unit): ULongArray {
    return apply { for (element in this)
        action(element) }
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.onEach(action: (UByte) -> Unit): UByteArray {
    return apply { for (element in this)
        action(element) }
}

Since Kotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.onEach(action: (UShort) -> Unit): UShortArray {
    return apply { for (element in this)
        action(element) }
}
action(element) 

Performs the given [action] on each element and returns the array itself afterwards.

Performs the given [action] on each element and returns the array itself afterwards.

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

@param [action] function that takes the index of an element and the element itself and performs the action on the element.

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

@param [action] function that takes the index of an element and the element itself and performs the action on the element.

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

@param [action] function that takes the index of an element and the element itself and performs the action on the element.

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

Throws an exception if this array is empty. If the array can be empty in an expected way, please use [reduceOrNull] instead. It returns `null` when its receiver is empty.

@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

Throws an exception if this array is empty. If the array can be empty in an expected way, please use [reduceOrNull] instead. It returns `null` when its receiver is empty.
accumulator

```kotlin
*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.reduce(operation: (acc: UShort, UShort) -> UShort): UShort {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.reduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): UShort {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

*\n
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty()) throw
    UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

Accumulates value starting with the first element and applying [operation] from left to right, to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
element, current accumulator value and the element itself, \n * and calculates the next accumulator value, \n * \n * @sample samples.collections.Collections.Aggregates.reduce

*\n@SinceKotlin("1.3\")\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(acc, accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes the index of an element, current accumulator value
 * and the element itself, \n * and calculates the next accumulator value, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reduceIndexedOrNull(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt?
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(acc, accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes the index of an element, current accumulator value
 * and the element itself, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.reduceIndexedOrNull(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong?
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(acc, accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes current accumulator value and an element, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceIndexedOrNull(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte?
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(acc, accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes current accumulator value and an element, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(acc, accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes current accumulator value and an element, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
UIntArray.reduceOrNull(operation: (acc: UInt, Uint) -> UInt): UInt?
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes current accumulator value and an element, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte?
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the first element
 * and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param \[operation\] function that takes current accumulator value and an element, \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n@SinceKotlin("1.4\")\n@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly\npublic inline fun
{ \n    if (isEmpty()) \n        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) \n        accumulator = operation(accumulator, this[index])
    return accumulator
}
element.\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes current accumulator value and an element.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceOrNull\n
\n*\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
\npublic inline fun ULongArray.reduceOrNull(operation: (acc: ULong, ULong) -> ULong): ULong? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)
        \n            accumulator = operation(accumulator, this[index])
    \n* \n* Accumulates value starting with the first element and applying [operation] from left to right\n* \n* to current accumulator value and each element.\n* \n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes current accumulator value and an element.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceOrNull\n
\n*\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
\npublic inline fun UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)
        \n            accumulator = operation(accumulator, this[index])
    \n* \n* Accumulates value starting with the first element and applying [operation] from left to right\n* \n* to current accumulator value and each element.\n* \n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes current accumulator value and an element.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceOrNull\n
\n*\n@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)
        \n            accumulator = operation(accumulator, this[index])
    \n* \n* Accumulates value starting with the first element and applying [operation] from left to right\n* \n* to current accumulator value and each element.\n* \n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes current accumulator value and an element.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceOrNull\n
\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
\npublic inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    \n        while (index >= 0)
            \n                accumulator = operation(get(index--), accumulator)
    \n* \n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n* \n* please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n* \n* @param [operation] function that takes an element and current accumulator value.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRight\n
\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
\npublic inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    \n        while (index >= 0)
            \n                accumulator = operation(get(index--), accumulator)
    \n* \n* Accumulates value starting with the last element and applying [operation] from right to left\n* \n* to each element and current accumulator value.\n* \n* \n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n* \n* please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n* \n* @param [operation] function that takes an element and current accumulator value.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRight\n
\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n
\npublic inline fun ULonArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    \n        while (index >= 0)
            \n                accumulator = operation(get(index--), accumulator)
    \n* \n* Accumulates value starting with the last element and applying [operation] from right to left\n* \n* to each element and current accumulator value.\n* \n* \n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n* \n* please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n* \n* @param [operation] function that takes an element and current accumulator value.\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRight\n

Open Source Used in DNAC Platform Arica 8058
```kotlin
/*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {\n    var index = lastIndex\n    if (index < 0) throw UnsupportedOperation(expected("Empty array can't be reduced.")\n    var accumulator = get(index--)\n    while (index != 0) {\n        accumulator = operation(accumulator, get(index--))\n    }\n    return accumulator}\n}**\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {\n    var index = lastIndex\n    if (index < 0) throw UnsupportedOperationException(expected("Empty array can't be reduced.")\n    var accumulator = get(index--)\n    while (index != 0) {\n        accumulator = operation(accumulator, get(index--))\n    }\n    return accumulator}\n}
```

```kotlin
/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * left\n * to each element and current accumulator value.\n * \n * Throws an exception if this array is empty. If the array can be empty in an expected way,\n * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRight
 */
```
expected way. * * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * @sample samples.collections.Collections.Aggregates.reduceRight

*A@SinceKotlin("1.3")@ExperimentalUnsignedTypes
*public inline fun
*UShortArray.reduceRightIndexedOrNull(operation: (index: Int, UShort, acc: UShort) -> UShort): UShort { * var index = lastIndex * if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.") * var accumulator = get(--index) * while (index >= 0) { * accumulator = operation(index, get(index), accumulator) * --index } * return accumulator * * * * * * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. * * * * * * Returns `null` if the array is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * * * * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

*A@SinceKotlin("1.4")@ExperimentalUnsignedTypes
*public inline fun
*UIntArray.reduceRightIndexedOrNull(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt? { * var index = lastIndex * if (index < 0) return null * var accumulator = get(--index) * while (index >= 0) { * accumulator = operation(index, get(index), accumulator) * --index } * return accumulator * * * * * * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. * * * * * * Returns `null` if the array is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * * * * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

*A@SinceKotlin("1.4")@ExperimentalUnsignedTypes
*public inline fun
*ULongArray.reduceRightIndexedOrNull(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong? { * var index = lastIndex * if (index < 0) return null * var accumulator = get(--index) * while (index >= 0) { * accumulator = operation(index, get(index), accumulator) * --index } * return accumulator * * * * * * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. * * * * * * Returns `null` if the array is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * * * * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

*A@SinceKotlin("1.4")@ExperimentalUnsignedTypes
*public inline fun
*UByteArray.reduceRightIndexedOrNull(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte? { * var index = lastIndex * if (index < 0) return null * var accumulator = get(--index) * while (index >= 0) { * accumulator = operation(index, get(index), accumulator) * --index } * return accumulator * * * * * * Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. * * * * * * Returns `null` if the array is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * * * * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

*A@SinceKotlin("1.4")@ExperimentalUnsignedTypes
*public inline fun
*UShortArray.reduceRightIndexedOrNull(operation: (index: Int, UShort, acc: UShort) -> UShort): UShort? { * var index = lastIndex * if (index < 0) return null * var accumulator = get(--index) * while (index >= 0) { * accumulator = operation(index, get(index), accumulator) * --index } * return accumulator * * * * * * Accumulates value starting with the last element and applying [operation] from right to left to each element and its index in the original array and current accumulator value. * * * * * * Returns `null` if the array is empty. * * * * @param [operation] function that takes the index of an element, the element itself and current accumulator value, * * and calculates the next accumulator value. * * * * * * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

Open Source Used In DNAC Platform Arica  8060
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightOrNull(operation: (UInt, acc: UInt) -> UInt): UInt? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param 
 * @param 
 * @sample 
 * @param 
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightOrNull(operation: (ULong, acc: ULong) -> ULong): ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param 
 * @param 
 * @sample 
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.runningFold(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right * to each element and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; * otherwise it would affect the
 * previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.runningFold(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        var accumulator = initial
        for (element in this) {
            accumulator = operation(accumulator, element)
            add(accumulator)
        }
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right * to each element and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; * otherwise it would affect the
 * previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        var accumulator = initial
        for (element in this) {
            accumulator = operation(accumulator, element)
            add(accumulator)
        }
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right * to each element, its index in the original array and current accumulator value
 * that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; * otherwise it would affect the
 * previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        var accumulator = initial
        for (index in indices) {
            accumulator = operation(index, accumulator, this[index])
            add(accumulator)
        }
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; * otherwise it would affect the
 * previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        var accumulator = initial
        for (index in indices) {
            accumulator = operation(index, accumulator, this[index])
            add(accumulator)
        }
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that ‘acc’ value passed to [operation] function should not be mutated; * otherwise it would affect the
 * previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply {
        var accumulator = initial
        for (index in indices) {
            accumulator = operation(index, accumulator, this[index])
            add(accumulator)
        }
    }
    return result
}
(isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.\n * @param [initial] value\n * @return List\n */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.runningFoldIndexed(initial: R, operation: (Int, acc: R, UShort) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * @param [initial] value\n * @return List\n */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.runningReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<UInt>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * @param [initial] value\n * @return List\n */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.runningReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<ULong>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this array.\n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n * @param [initial] value\n * @return List\n */
@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.runningReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<UByte>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result
}
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}

@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun UIntArray.runningReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}

@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun ULongArray.runningReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator)\n        for (index in 1 until size) {\n            accumulator = operation(index, accumulator, this[index])\n            result.add(accumulator)\n        }\n        return result\n    }
}

@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun UByteArray.runningReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply {
        add(accumulator)\n        for (index in 1 until size) {\n            accumulator = operation(index, accumulator, this[index])\n            result.add(accumulator)\n        }\n        return result\n    }
}

@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator)\n        for (index in 1 until size) {\n            accumulator = operation(index, accumulator, this[index])\n            result.add(accumulator)\n        }\n        return result\n    }
}
values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list. Otherwise it would affect the previous value in resulting list.
R, ULong) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

@Deprecated("Use runningReduce instead.", ReplaceWith("runningReduce(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.scanReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {
    return runningReduce(operation)
}

@Deprecated("Use runningReduceIndexed instead.", ReplaceWith("runningReduceIndexed(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.scanReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> {
    return runningReduceIndexed(operation)
}

Open Source Used In DNAC Platform Arica  8066
DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.scanReduceIndexed(operation: (Int, acc: UShort) -> UShort) {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumBy(selector: (ULong) -> ULong): ULong {
    var sum: ULong = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumBy(selector: (UByte) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumBy(selector: (UShort) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumByDouble(selector: (UInt) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumOf(selector: (UInt) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumOf(selector: (ULong) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumOf(selector: (UByte) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UByteArray.sumOf(selector: (UByte) -> Double): Double { var sum: Double = 0.toDouble() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UShortArray.sumOf(selector: (UShort) -> Double): Double { var sum: Double = 0.toDouble() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UIntArray.sumOf(selector: (UInt) -> Int): Int { var sum: Int = 0.toInt() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UByteArray.sumOf(selector: (UByte) -> Int): Int { var sum: Int = 0.toInt() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UShortArray.sumOf(selector: (UShort) -> Int): Int { var sum: Int = 0.toInt() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun ULongArray.sumOf(selector: (ULong) -> Long): Long { var sum: Long = 0.toLong() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.

*An@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@Suppress("INAPPLICABLE_JVM_NAME")@kotlin.jvm.JvmName("sumOfDouble")
@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun UIntArray.sumOf(selector: (UInt) -> Long): Long { var sum: Long = 0.toLong() for (element in this) { sum += selector(element) return sum } } * Returns the sum of all values produced by [selector] function applied to each element in the array.
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfLong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumOf(selector: (UByte) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfLong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumOf(selector: (UShort) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumOf(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumOf(selector: (ULong) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

---

Open Source Used In DNAC Platform Arica 8069
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumOf(selector: (UByte) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@Suppress("INAPPLICABLE_JVM_NAME")
@kotlin.jvm.JvmName("sumOfULong")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumOf(selector: (UShort) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> UByteArray.zip(other: Array<out R>): List<Pair<UByte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> UShortArray.zip(other: Array<out R>): List<Pair<UShort, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index.
 *
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

---

Open Source Used In DNAC Platform Arica 8070
with the same index

* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.

* The returned list has length of the shortest collection.
UShortArray.zip(other: Iterable<R>, transform: (a: UShort, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterable

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongArray.zip(other: ULongArray): List<Pair<ULong, ULong>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UByteArray.zip(other: UByteArray): List<Pair<UByte, UByte>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
    return zip(other) { t1, t2 -> t1 to t2 }
}
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UCollectionsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an array of UByte containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UByte>.toUByteArray(): UByteArray {
    val result = UByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of UInt containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UInt>.toUIntArray(): UIntArray {
    val result = UIntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of ULong containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<ULong>.toULongArray(): ULongArray {
    val result = ULongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of UShort containing all of the
 * elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UShort>.toUShortArray(): UShortArray {
    val result = UShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }

    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfLong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }

    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }

    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }

    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.sum(): UInt {
    return storage.sum().toUInt()
}

/**
 * Returns the sum of all elements in the
 * array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.sum(): ULong {
    return storage.sum().toULong()
}

/**
 * Returns the sum of all elements in the
 * array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.sum(): UInt {
    return sumBy { it.toUInt() }
}

/**
 * Returns the sum of all elements in the
 * array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.sum(): UInt {
    return sumBy { it.toUInt() }
}
Iterable<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

fun maxOf(a: UInt, vararg other: UInt): UInt {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

fun maxOf(a: ULong, vararg other: ULong): ULong {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

fun minOf(a: UByte, vararg other: UByte): UByte {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

fun minOf(a: UShort, vararg other: UShort): UShort {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

fun minOf(a: ULong, vararg other: ULong): ULong {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

fun minOf(a: UInt, vararg other: UInt): UInt {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}
minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(minOf(a, minOf(b, c)), a)
}

    return minOf(minOf(a, minOf(b, c)), a)
}

minOf(a: UInt, vararg other: UInt): UInt {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

minOf(a: ULong, vararg other: ULong): ULong {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

minOf(a: UByte, vararg other: UByte): UByte {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

minOf(a: UShort, vararg other: UShort): UShort {
    var min = a
    for (e in other) min = minOf(min, e)
    return min
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("URangesKt")

package kotlin.ranges

import java.util.*

/**
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntRange.random(): UInt {
    return random(random())
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntRange.random(random: Random): UInt {
    try {
        return random.nextUInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public fun UIntRange.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return random.nextUInt(this)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public fun ULongRange.random(): ULong {
    return random(random())
}

/**
 * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public fun ULongRange.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return random.nextULong(this)
}

/**
 * Returns `true` if this range contains the specified [element].
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline operator fun UIntRange.contains(element: UInt): Boolean

**UIntRange.contains(element: UInt?): Boolean**

```kotlin
UIntRange.contains(element: UInt?): Boolean {
    return element != null && contains(element)
}
```

```kotlin
/**
 * Returns `true` if this range contains the specified [element].
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline operator fun UIntRange.contains(element: UInt?): Boolean {
    return element != null && contains(element)
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: UByte): Boolean {
    return contains(value.toUInt())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: UByte): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: UInt): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: ULong): Boolean {
    return (value shr UInt.SIZE_BITS) == 0uL && contains(value.toUInt())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: UShort): Boolean {
    return contains(value.toUInt())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: UShort): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to this value.
 * If the [to] value is greater than this value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to this value.
 * If the [to] value is greater than this value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1, this.step, this.step)
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to this value.
 * If the [to] value is greater than this value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULong.downTo(to: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(this, to, -1, this.step, this.step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongProgression.reversed(): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UIntProgression.step(step: Int): UIntProgression {
    checkStepIsPositive(step > 0, step)
    return UIntProgression.fromClosedRange(first, last, step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongProgression.step(step: Long): ULongProgression {
    checkStepIsPositive(step > 0, step)
    return ULongProgression.fromClosedRange(first, last, step)
}
```
this value up to but excluding the specified [to] value.\n * @return a range from this value up to but excluding the specified [to] value.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun
UByte.until(to: UByte): UIntRange {
    if (to <= UByte.MIN_VALUE) return UIntRange.EMPTY\n    return this.toUInt() .. (to - 1u).toUInt()\n
/**
 * Returns a range from this value up to but excluding the specified [to] value.\n * @return a range from this value up to but excluding the specified [to] value.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UInt.until(to: UInt): UIntRange {
    if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY\n    return this .. (to - 1u).toUInt()\n
/**
 * Returns a range from this value up to but excluding the specified [to] value.\n * @return a range from this value up to but excluding the specified [to] value.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULong.until(to: ULong): ULongRange {
    if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY\n    return this .. (to - 1u).toULong()\n
/**
 * Ensures that this value is not less than or equal to the specified [minimumValue].\n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtLeast(minimumValue: UInt): UInt {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than or equal to the specified [minimumValue].\n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtLeast(minimumValue: ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than or equal to the specified [minimumValue].\n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtLeast(minimumValue: UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than or equal to the specified [minimumValue].\n * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtLeast(minimumValue: UShort): UShort {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].\n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtMost(maximumValue: UInt): UInt {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].\n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtMost(maximumValue: ULong): ULong {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].\n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].\n * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.\n * @throws IllegalArgumentException if the [from] value is greater than or equal to the [to] value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtMost(maximumValue: UShort): UShort {
    return if (this > maximumValue) maximumValue else this
}
UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue]; otherwise.
 * @sinceKotlin("1.3")
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..<maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sinceKotlin("1.3")
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..<maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sinceKotlin("1.3")
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..<maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sinceKotlin("1.3")
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..<maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sinceKotlin("1.3")
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
fun ULong.coerceIn(range: ClosedRange<ULong>): ULong {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<ULong>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

val error: Throwable
    get() = when {
    this is Error -> this
    this is Exception -> this
    this is RuntimeException -> this
    this is IllegalArgumentException -> this
    this is IllegalStateException -> this
    this is IndexOutOfBoundsException -> this
    this is ConcurrentModificationException -> this
    this is UnsupportedOperationException -> this
    this is NumberFormatException -> this
    this is NullPointerException -> this
    this is ClassCastException -> this
    this is AssertionError -> this
    else -> throw this
    }

expect open class Error : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}
expect open class Exception : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}
expect open class RuntimeException : Exception {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}
expect open class IllegalArgumentException : RuntimeException {
    constructor()
    constructor(message: Any?)
}
expect open class IllegalStateException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}
expect open class IndexOutOfBoundsException : RuntimeException {
    constructor()
    constructor(message: String?)
}
expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
    @Deprecated("The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
    constructor(message: String?, cause: Throwable?)
    @Deprecated("The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
    constructor(cause: Throwable?)
}
expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}
expect open class NumberFormatException : IllegalArgumentException {
    constructor()
    constructor(message: String?)
}
expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}
expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}
expect open class AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

NoSuchElementException : RuntimeException {
    constructor()
    constructor(message: String?)
}

@SinceKotlin("1.3")
public expect open class ArithmeticException : RuntimeException {
    constructor()
    constructor(message: String?)
}

@Deprecated("This exception type is not supposed to be thrown or caught in common code and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
public expect open class NoWhenBranchMatchedException : RuntimeException {
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

@Deprecated("This exception type is not supposed to be thrown or caught in common code and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
public expect class UninitializedPropertyAccessException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

/**
 * Throws after invocation of a function or property that was expected to return `Nothing`,
 * but returned something instead.
 */
@SinceKotlin("1.4")
@PublishedApi
internal class KotlinNothingValueException : RuntimeException {
    constructor() : super()
    constructor(message: String?) : super(message)
    constructor(message: String?, cause: Throwable?) : super(message, cause)
    constructor(cause: Throwable?) : super(cause)
}

/**
 * Returns the detailed description of this throwable
 * with its stack trace.
 */
@SinceKotlin("1.4")
public expect fun Throwable.stackTraceToString(): String

/**
 * Prints the [detailed description][Throwable.stackTraceToString]
 * of this throwable to the standard output or standard error output.
 */
@SinceKotlin("1.4")
public expect fun Throwable.printStackTrace(): Unit

/**
 * When supported by the platform, adds the specified exception to the
 * list of exceptions that were suppressed in order to deliver this exception.
 * Returns a list of all exceptions that were
 * suppressed in order to deliver this exception.
 * The list can be empty.
 * - if no exceptions were suppressed;
 * - if the platform doesn't support suppressed exceptions;
 * - if this [Throwable] instance has disabled the suppression.
 */
@SinceKotlin("1.4")
public expect fun Throwable.addSuppressed(exception: Throwable)

/**
 * Returns a list of all exceptions that were
 * suppressed in order to deliver this exception.
 * The list can be empty:
 * - if no exceptions were suppressed;
 * - if the platform doesn't support suppressed exceptions.
 * Usage of such annotations will be reported as warnings unless an explicit opt-in with
 * the [OptIn] annotation, e.g.
 * `@OptIn(ExperimentalJsExport::class)`,
 * is given.
 */
@Experimental
@RequiresOptIn
@OptIn(ExperimentalJsExport::class)
fun channel(): String

// Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

No open source used in DNAC Platform Arica 8080
Open Source Used In DNAC Platform Arica 8081

The classes in this interface support only read-only access to the collection.

The collection is covariant in its element type.

public interface List<out E> : Collection<E> {

    // Positional Access

    contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations

    replaceAll(elements: Collection<@UnsafeVariance E>): Boolean
    clear(): Unit
    retainAll(elements: Collection<E>): Boolean
    removeAll(elements: Collection<E>): Boolean

    // Query Operations

    override val size: Int
    override fun isEmpty(): Boolean
    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Returns the element at the specified index in the list.
     */
    public operator fun get(index: Int): E

    override fun iterator(): Iterator<E>

    // Bulk Modification Operations
    /**
     * Adds all of the elements of the specified collection
     * to this collection.
     *
     * @return `true` if any of the specified elements was added to the collection, `false` if
     * the collection was not modified.
     */
    public fun addAll(elements: Collection<E>): Boolean

    /**
     * Adds the specified element to the collection.
     *
     * @return `true` if the element has been added, `false` if the collection does not support
     * duplicates and the element is already contained in the collection.
     */
    public fun add(element: E): Boolean

    /**
     * Removes all of the collection's elements that are also contained in the specified collection.
     * @return `true` if any of the specified elements was removed from the collection, `false` if
     * the collection was not modified.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Removes all from this collection.
     * @return `true` if any element was removed from the collection, `false` if the collection
     * was not modified.
     */
    public fun remove(element: E): Boolean

    /**
     * Removes a single instance of the specified element from this collection, if it is present.
     * @return `true` if the element has been successfully removed, `false` if it was not present.
     */
    public fun removeAt(index: Int): Boolean
}

/**
 * A generic collection of elements. Methods in this interface support only read-only access to
 * the collection;
 * read/write access is supported through the [MutableList] interface.
 * @param T the type of elements contained in the collection.
 * @param E the type of elements supported during iteration.
 */
public interface Collection<out E> : Iterable<E> {
    // Query Operations

    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public operator fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations

    removeAll(elements: Collection<E>): Boolean
    clear(): Unit
    retainAll(elements: Collection<E>): Boolean
    removeAll(elements: Collection<E>): Boolean

    /**
     * Removes all of this collection's elements that are also contained in the specified collection.
     * @return `true` if any of the specified elements was removed from the collection, `false` if
     * the collection was not modified.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Returns the size of the collection.
     */
    public val size: Int

    /**
     * Returns
     * `true` if the collection is empty (contains no elements), `false` otherwise.
     */
    public fun isEmpty(): Boolean

    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public operator fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations

    /**
     * Removes all of the elements of the specified collection
     * from this collection.
     * @param T the type of element
     * @param E the type of elements contained in the list.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Removes all from this list.
     * @param T the type of element
     * @param E the type of elements contained in the list.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Returns an iterator over the elements of this sequence that supports removing elements
during iteration.
     */
    override fun iterator(): Iterator<T>

    /**
     * Returns
     * `true` if the collection is empty (contains no elements), `false` otherwise.
     */
    public fun isEmpty(): Boolean

    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public operator fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations

    /**
     * Removes all of the elements of the specified collection
     * from this collection.
     * @param T the type of element
     * @param E the type of elements contained in the list.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Returns an iterator over the elements of this sequence that supports removing elements
during iteration.
     */
    override fun iterator(): Iterator<T>

    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public operator fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations

    /**
     * Removes all of the elements of the specified collection
     * from this collection.
     * @param T the type of element
     * @param E the type of elements contained in the list.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Returns an iterator over the elements of this sequence that supports removing elements
during iteration.
     */
    override fun iterator(): Iterator<T>

    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public operator fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations
public abstract class BooleanIterator : Iterator<Boolean> {
    override final fun next() = nextBoolean()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextBoolean(): Boolean
}

/** An iterator over a sequence of values of type `Boolean`. */
public abstract class BooleanIterator : Iterator<Boolean> {
    override final fun next() = nextBoolean()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextBoolean(): Boolean
}

/** An iterator over a sequence of values of type `Long`. */
public abstract class LongIterator : Iterator<Long> {
    override final fun next() = nextLong()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextLong(): Long
}

/** An iterator over a sequence of values of type `Long`. */
public abstract class LongIterator : Iterator<Long> {
    override final fun next() = nextLong()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextLong(): Long
}

/** An iterator over a sequence of values of type `Char`. */
public abstract class CharIterator : Iterator<Char> {
    private val finalElement = last.toInt()
    private var hasNext: Boolean = if (step > 0) first <= finalElement
        else last >= finalElement
    private var next = if (hasNext) first.toInt() else finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextChar(): Char {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        }
        else {
            next += step
        }
        return value.toChar()
    }
}

/** An iterator over a sequence of values of type `Char`. */
public abstract class CharIterator : Iterator<Char> {
    private val finalElement = last.toInt()
    private var hasNext: Boolean = if (step > 0) first <= finalElement
        else last >= finalElement
    private var next = if (hasNext) first.toInt() else finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextChar(): Char {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        }
        else {
            next += step
        }
        return value.toChar()
    }
}

// Auto-generated file. DO NOT EDIT!

package kotlin.ranges

/**
 * A progression of values of type `Char`.
 * @property step the number by which the value is incremented on each step.
 * @property finalElement the last value in the progression.
 */
public class CharProgression : Progression<Char> {
    internal constructor
    (
            start: Char,
            endInclusive: Char,
            step: Int
    ) : super(start, endInclusive, step)

    override fun iterator(): CharIterator = CharProgressionIterator

    companion object {
        val NO SUCH ELEMENT EXCEPTION: ClassCastException
            = throw kotlin.NoSuchElementException()
        val MIN_VALUE: Int = Int.MIN_VALUE
        val MAX_VALUE: Int = Int.MAX_VALUE
    }

    override fun toString(): String = if (step == 0) "" else "from $start to $endInclusive step $step"
}

<CharProgression>

/**
 * A progression of values of type `Char`.
 * @property step the number by which the value is incremented on each step.
 * @property finalElement the last value in the progression.
 */
public class CharProgression : Progression<Char> {
    internal constructor
    (
            start: Char,
            endInclusive: Char,
            step: Int
    ) : super(start, endInclusive, step)

    override fun iterator(): CharIterator = CharProgressionIterator

    companion object {
        val NO SUCH ELEMENT EXCEPTION: ClassCastException
            = throw kotlin.NoSuchElementException()
        val MIN_VALUE: Int = Int.MIN_VALUE
        val MAX_VALUE: Int = Int.MAX_VALUE
    }

    override fun toString(): String = if (step == 0) "" else "from $start to $endInclusive step $step"
}

<CharProgression>
other is CharProgression & & (isEmpty()) & & other.isEmpty()) || ]n first == other.first & & last == other.last & & step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step)\n\n override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $step"\n\n companion object {\n /**\n * Creates CharProgression within the specified bounds of a closed range.\n * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n * In order to go backwards the [step] must be negative.\n * [step] must be greater than 'Int.MIN_VALUE' and not equal to zero.\n */\n\n public fun fromClosedRange(rangeStart: Char, rangeEnd: Char, step: Int): CharProgression =\n CharProgression(rangeStart, rangeEnd, step)\n ]n]n/*\n * A progression of values of type 'Long'.\n */\n\n public open class LongProgression internal constructor\n (\n start: Long,\n endInclusive: Long,\n step: Long\n ) :\n Iterable<Long> {\n init {\n if (step == 0L) throw kotlin.IllegalArgumentException("Step must be non-zero.");\n other is LongProgression & & (isEmpty() & & other.isEmpty()) || ]n first == other.first & & last == other.last & & step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * first + last) + step)\n\n override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $step"\n\n companion object {\n /**\n * Checks if the progression is empty.\n */\n\n public val first: Long = start\n
 /**\n * The first element in the progression.\n */\n\n public val last: Long = getProgressionLastElement(start.toLong(), endInclusive.toLong(), step).toLong()\n
 /**\n * The step of the progression.\n */\n\n public val step: Long = step\n
 /**\n * The last element in the progression.\n */\n\n internal constructor\n (\n start: Long,\n endInclusive: Long,\n step: Long\n ) :\n Iterable<Long> {\n init {\n if (step == 0L) throw kotlin.IllegalArgumentException("Step must be non-zero.");\n other is LongProgression & & (isEmpty() & & other.isEmpty()) || ]n first == other.first & & last == other.last & & step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * first + last) + step)\n\n override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $step"\n\n companion object {\n /**\n * Creates LongProgression within the specified bounds of a closed range.\n * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n * In order to go backwards the [step] must be negative.\n * [step] must be greater than 'Long.MIN_VALUE' and not equal to zero.\n */\n\n public fun fromClosedRange(rangeStart: Long, rangeEnd: Long, step: Long): LongProgression =\n LongProgression(rangeStart, rangeEnd, step)\n ]n]n/*\n * A progression of values of type 'Int'.\n */\n\n public open class IntProgression internal constructor\n (\n start: Int,\n endInclusive: Int,\n step: Int\n ) :\n Iterable<Int> {\n init {\n if (step == 0) throw kotlin.IllegalArgumentException("Step must be non-zero.");\n other is IntProgression & & (isEmpty() & & other.isEmpty()) || ]n first == other.first & & last == other.last & & step == other.step)\n\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * (31 * first + last) + step)\n\n override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $step"\n\n companion object {\n /**\n * Checks if the progression is empty.\n */\n\n public val first: Int = start\n
 /**\n * The first element in the progression.\n */\n\n public val last: Int =\n Long.MIN_VALUE to avoid overflow on negation.")\n    }

    /**
     * The step
     */
    public val step: Int = step

    override fun iterator(): IntIterator =
    LongProgressionIterator(first, last, step)

    /** Checks if the progression is empty. */
    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
    if (isEmpty()) false else\n    other is LongProgression && (isEmpty() && other.isEmpty() ||
    first == other.first && last == other.last &&
    isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun hashCode(): Int =
    if (isEmpty()) -1 else (31 * (31 * first + last) + step)
Open Source Used In DNAC Platform Arica  8086

documentation](https://kotlinlang.org/docs/reference/ranges.html) for more information.

```java
public interface ClosedRange<T: Comparable<T>> {
    /**
     * The minimum value in the range.
     */
    public val start: T

    /**
     * The maximum value in the range (inclusive).
     */
    public val endInclusive: T

    /**
     * Checks whether the specified [value] belongs to the range.
     */
    operator fun contains(value: T): Boolean = value >= start && value <= endInclusive

    /**
     * Checks whether the range is empty.
     */
    public fun isEmpty(): Boolean = start > endInclusive
}
```

```java
package kotlin

/**
 * The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java.
 */
public object Unit {
    override fun toString() = "kotlin.Unit"
}
```

```java
package kotlin.annotation

import kotlin.annotation.AnnotationTarget.*

/**
 * Contains the list of code elements which are the possible annotation targets
 */
public enum class AnnotationTarget {
    /**
     * Class, interface or object, annotation class is also included
     */
    CLASS,
    /**
     * Annotation class only
     */
    ANNOTATION_CLASS,
    /**
     * Generic type parameter (unsupported yet)
     */
    TYPE_PARAMETER,
    /**
     * Property
     */
    PROPERTY,
    /**
     * Field, including property's backing field
     */
    FIELD,
    /**
     * Local variable
     */
    LOCAL_VARIABLE,
    /**
     * Value parameter of a function or a constructor
     */
    VALUE_PARAMETER,
    /**
     * Constructor only (primary or secondary)
     */
    CONSTRUCTOR,
    /**
     * Function (constructors are not included)
     */
    FUNCTION,
    /**
     * Property getter only
     */
    PROPERTY_GETTER,
    /**
     * Property setter only
     */
    PROPERTY_SETTER,
    /**
     * Type usage
     */
    TYPE,
    /**
     * Any expression
     */
    EXPRESSION,
    /**
     * File
     */
    FILE,
    /**
     * Type alias
     */
    @SinceKotlin("1.1")
    TYPEALIAS

    /**
     * Contains the list of possible annotation targets
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @MustBeDocumented
    public annotation class Target(vararg allowedTargets: AnnotationTarget)

    /**
     * Determines how an annotation is stored in binary output.
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    public annotation class Retention(val value: AnnotationRetention = AnnotationRetention.RUNTIME)

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    public annotation class Repeatable

    /**
     * Determines that an annotation is a part of public API and therefore should be included in the generated documentation for the element to which the annotation is applied.
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable

    /**
     * Determines that an annotation is applicable twice or more on a single code element
     */
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    @Repeatable
    @Target(AnnotationTarget.ANNOTATION_CLASS)
    @Retention(AnnotationRetention.RUNTIME)
    @MustBeDocumented
    public annotation class Repeatable
@Target(AnnotationTarget.ANNOTATION_CLASS)
@Retention(AnnotationRetention.BINARY)
@internal annotation class PureReifiable

/**
 * Specifies that the corresponding built-in method exists depending on platform.
 * Current implementation for JVM looks whether method with same JVM descriptor exists in the module JDK.
 * For example MutableMap.remove(K, V) available only if corresponding method 'java/util/Map.remove(Ljava/lang/Object;Ljava/lang/Object;)Z' is defined in JDK (i.e. for major versions >= 8)
 */
@Target(AnnotationTarget.TYPE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
internal annotation class PureReifiable

/**
 * Specifies that the corresponding built-in method exists depending on platform.
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

@JsName("arrayIterator")
internal fun arrayIterator(array: dynamic, type: String?) = when (type) {
    null -> {
        val arr: Array<dynamic> = array
        object : Iterator<dynamic> {
            var index = 0
            override fun hasNext() = index < arr.size
            override fun next() = if (index < arr.size) arr[index++]
        }
    }
    "BooleanArray" -> booleanArrayIterator(array)
    "ByteArray" -> byteArrayIterator(array)
    "ShortArray" -> shortArrayIterator(array)
    "CharArray" -> charArrayIterator(array)
    else -> throw NoSuchElementException(""Array does not have iterator for type $type""
        )
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// a mod b (in arithmetical sense)
private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

// (a - b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int {
    return mod(mod(a, c) - mod(b, c), c)
}

private fun differenceModulo(a: Long, b: Long, c: Long): Long {
    return mod(mod(a, c) - mod(b, c), c)
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

// a mod b (in arithmetical sense)
private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

// (a - b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int {
    return mod(mod(a, c) - mod(b, c), c)
}

private fun differenceModulo(a: Long, b: Long, c: Long): Long {
    return mod(mod(a, c) - mod(b, c), c)
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range from [start] to [end] case of a positive [step], or from [end] to [start] case of a negative [step].
 * The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end final element of the progression
 * @param step difference of successive elements in the progression
 * @return the final element of the progression
 */
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when
{ step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
 step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
 else -> throw kotlin.IllegalArgumentException("Step is zero.")
}
intArrayIterator(array) → longArrayIterator(array) → floatArrayIterator(array) → doubleArrayIterator(array) else → throw IllegalStateException("Unsupported type argument for arrayIterator:
Sype\n")

private fun booleanArrayIterator(array: BooleanArray) =
object : BooleanIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextBoolean() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun byteArrayIterator(array: ByteArray) = object : ByteIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextByte() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun shortArrayIterator(array: ShortArray) = object : ShortIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextShort() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun charArrayIterator(array: CharArray) = object : CharIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextChar() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun intArrayIterator(array: IntArray) = object : IntIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextInt() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun floatArrayIterator(array: FloatArray) = object : FloatIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextFloat() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun doubleArrayIterator(array: DoubleArray) = object : DoubleIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextDouble() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun longArrayIterator(array: LongArray) = object : LongIterator() {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextLong() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

private fun subSequence(c: CharSequence, startIndex: Int, endIndex: Int): CharSequence {
    if (c is String) {
        return c.substring(startIndex, endIndex)
    } else {
        return c.asDynamic().`subSequence_vux9f0$`(startIndex, endIndex)
    }
}

private fun captureStack(baseClass: JsClass<in Throwable>, instance: Throwable) {
    if (js("Error").captureStackTrace) {
        // Using uncropped stack traces due to KT-37563.
        // Precise stack traces are implemented in JS IR compiler and stdlib
        js("Error").captureStackTrace(instance);
    } else {
        instance.asDynamic().stack = js("new Error()").stack;
    }
}

private fun newThrowable(message: String?, cause: Throwable?): Throwable {
    val throwable = js("new Error()")
    throwable.message = if (jsTypeOf(message) == "undefined") {
        if (cause != null) cause.toString() else null
    } else {
        message
    }
    throwable.cause = cause
    throwable.name = "Throwable"
    return throwable
}

@kotlin.internal.InlineOnly
internal inline fun <T> concat(args: Array<T>): T {
    val typed = js("Array")(*args)
}
Open Source Used In DNAC Platform Arica  8089

typed[i] = js(["\n"].slice.call(arr)\n) else \n    typed[i] = arr\n\n} return
js(["\n"].concat.apply(js(["\n"], typed))\n)\n
/**
 * Concat regular Array's and TypedArray's into an Array.
 */
@PublishedApi
@JsName("arrayConcat")
@Suppress("UNUSED_PARAMETER")
internal fun <T>
arrayConcat(a: T, b: T): T {
    return concat(js(["arguments"]))\n)\n
/**
 * Concat primitive arrays. Main use: prepare vararg arguments.\n * For compatibility with 1.1.0 the arguments may be a mixture of Array's and
TypedArray's.\n * If the first argument is TypedArray (Byte-, Short-, Char-, Int-, Float-, and DoubleArray)
returns a TypedArray, otherwise an Array.\n * If the first argument has the $type$ property (Boolean-, Char-, and
LongArray) copy its value to result.$type$.\n * If the first argument is a regular Array without the $type$ property
default to arrayConcat.\n */
@PublishedApi
@JsName("primitiveArrayConcat")
@Suppress("UNUSED_PARAMETER")
internal
fun <T> primitiveArrayConcat(a: T, b: T): T {
    val args: Array<T> = js("arguments")\n    if (a is Array<*> &&
a.asDynamic().$type$ === undefined) {
        return concat(args)\n    } else {
        var size = 0\n        for (i in
            args.indices) {
            size += args[i].asDynamic().length as Int\n        }\n        val result = js("new\n            a.constructor(size)")\n        kotlin.copyArrayType(a, result)\n        size = 0\n        for (i in
            args.indices) {
            val
                arr = args[i].asDynamic()\n            for (j in 0 until arr.length) {
                result[size++] = arr[j]\n            }\n        }\n        return result\n    }\n}

@JsName("booleanArrayOf")
internal fun booleanArrayOf() =
    withType("BooleanArray", js(["\n"].slice.call(arguments)))\n
@JsName("charArrayOf")
internal fun charArrayOf() =
    withType("CharArray", js("new\n        Uint16Array(["\n"].slice.call(arguments))"))\n
@JsName("longArrayOf")
internal fun longArrayOf() =
    withType("LongArray", js(["\n"].slice.call(arguments)))\n
@JsName("withType")
@kotlin.internal.InlineOnly
internal inline fun
    withType(type: String, array: dynamic): dynamic {
        array.$type$ = type\n        return array
    } // The arguments have to be slice'd here because of Rhino (see KT-16974)\n
package kotlin.js

/**
 * Function corresponding to JavaScript's `typeof` operator
 */
@kotlin.internal.InlineOnly
@Suppress("UNUSED_PARAMETER")
public inline fun jsTypeOf(a: Any?):
    String = js("typeof a")

@file:Suppress("UNUSED_PARAMETER", "NOTHING_TO_INLINE")

package kotlin

/**
 * Returns an empty array of the specified type [T].\n */
public inline fun <T> emptyArray(): Array<T> = js(["\n")\n
@library
public fun <T> arrayOf(vararg elements: T): Array<T> = definedExternally

@library
public fun doubleArrayOf(vararg elements: Double): DoubleArray = definedExternally

@library
public fun floatArrayOf(vararg elements: Float): FloatArray = definedExternally

@library
public fun longArrayOf(vararg elements: Long): LongArray =
    definedExternally\n
@library
public fun intArrayOf(vararg elements: Int): IntArray =
    definedExternally\n
@library
public fun charArrayOf(vararg elements: Char): CharArray =
    definedExternally\n
@library
public fun shortArrayOf(vararg elements: Short): ShortArray =
    definedExternally\n
@library
public fun byteArrayOf(vararg elements: Byte): ByteArray =
    definedExternally\n
@library
public fun booleanArrayOf(vararg elements: Boolean): BooleanArray =
    definedExternally\n
*\n
internal fun fillFrom(src: dynamic, dst: dynamic):
    dynamic {
        val srcLen: Int = src.length
        val dstLen: Int = dst.length\n        var index: Int = 0\n        while (index < srcLen && index < dstLen) {
            dst[index] = src[index++]\n        }\n        return dst\n    }\n
internal fun arrayCopyResize(source:
dynamic, newSize: Int, defaultValue: Any?): dynamic {
    val result = source.slice(0, newSize)
    copyArrayType(source, result)
    var index: Int = source.length
    if (newSize > index) {
        result.length = newSize
        while (index < newSize) result[index++] = defaultValue
    }
    return result
}

internal fun <T> arrayPlusCollection(array: dynamic, collection: Collection<T>): dynamic {
    val result = array.slice()
    result.length += collection.size
    copyArrayType(array, result)
    var index: Int = array.length
    for (element in collection) result[index++] = element
    return result
}

internal fun <T> fillFromCollection(dst: dynamic, startIndex: Int, collection: Collection<T>): dynamic {
    var index = startIndex
    for (element in collection) dst[index++] = element
    return dst
}

internal inline fun copyArrayType(from: dynamic, to: dynamic) {
    if (from.`$type$` !== undefined) {
        to.`$type$` = from.`$type$`
    }
}

internal inline fun jsIsType(obj: dynamic, jsClass: dynamic) = js("Kotlin").isType(obj, jsClass)
Any?
) : Serializable {
    
    // discovery
    /*
     * Returns `true` if this instance represents a successful outcome.
     * In this case [isFailure] returns `false`.
     */
    public val isSuccess: Boolean get() = value !is Failure
    /*
     * Returns `true` if this instance represents a failed outcome.
     * In this case [isSuccess] returns `false`.
     */
    public val isFailure: Boolean get() = value is Failure
    // value & exception retrieval
    /*
     * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or `null`
     * if it is [failure][Result.isFailure].
     */
    inline fun success(value: T): Result<T> =
        Result(value)
    /*
     * Returns the encapsulated [Throwable] exception if this instance represents [failure][isFailure] or `null`
     * if it is [success][isSuccess].
     */
    inline fun failure(exception: Throwable): Result<T> =
        Result(createFailure(exception))

    /**
     * Companion object for [Result] class that contains its constructor functions
     * [success] and [failure].
     */
    public companion object {
        /**
         * Returns a string `Success(v)` if this instance represents
         * [success][Result.isSuccess] where `v` is a string representation of the value or a string `Failure(x)` if
         * it is [failure][isFailure] where `x` is a string representation of the exception.
         */
        public fun exceptionOrNull(): Throwable? =
            when (value) {
                is Failure -> value.exception
                else -> null
            }
        /**
         * Returns `true` if this instance represents a successful
         * outcome.
         * In this case [isSuccess] returns `true`.
         */
        public val isSuccess: Boolean get() = value !is Failure
        // value
        /*
         * Returns `false` if this instance represents a failed outcome.
         * In this case [isSuccess] returns `false`.
         */
        public val isFailure: Boolean get() = value is Failure
    }

    // value & exception
    /**
     * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or `null`
     * if it is [failure][Result.isFailure].
     */
    inline fun getOrNull(): T? =
        when {
            isFailure -> null
            else -> value as T
        }
    /**
     * Returns the encapsulated [Throwable] exception if this instance represents [failure][isFailure] or `null`
     * if it is [success][isSuccess].
     */
    inline fun getOrThrow(): Throwable? =
        when (value) {
            is Failure -> null
            else -> value as T
        }

    /**
     * Returns a shorthand for `getOrElse { null }` (see [getOrElse]).
     */
    @InlineOnly
    @SinceKotlin("1.3")
    public inline fun <R, T :
    $$
    @Suppress("INAPPLICABLE_JVM_NAME")
    @InlineOnly
    @JvmName("success")
    public inline fun <T> success(value: T): Result<T> =
        Result(value)
    /*
     * Returns an instance that encapsulates the given [value] as successful value.
     */
    @Suppress("INAPPLICABLE_JVM_NAME")
    @InlineOnly
    @JvmName("failure")
    public inline fun <T> failure(exception: Throwable): Result<T> =
            Result(createFailure(exception))

    /**
     * Companion object for [Result]
     */
    public companion object {
        /**
         * Returns a string `Success(v)` if this instance represents
         * [success][Result.isSuccess] where `v` is a string representation of the value or a string `Failure(x)` if
         * it is [failure][isFailure] where `x` is a string representation of the exception.
         */
        public fun exceptionOrNull(): Throwable? =
            when (value) {
                is Failure -> value.exception
                else -> null
            }
        /**
         * Returns `true` if this instance represents a successful
         * outcome.
         * In this case [isSuccess] returns `true`.
         */
        public val isSuccess: Boolean get() = value !is Failure
        // value
        /*
         * Returns `false` if this instance represents a failed outcome.
         * In this case [isSuccess] returns `false`.
         */
        public val isFailure: Boolean get() = value is Failure
    }

    // value & exception
    /**
     * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or `null`
     * if it is [failure][Result.isFailure].
     */
    inline fun getOrNull(): T? =
        when {
            isFailure -> null
            else -> value as T
        }
    /**
     * Returns the encapsulated [Throwable] exception if this instance represents [failure][isFailure] or `null`
     * if it is [success][isSuccess].
     */
    inline fun getOrThrow(): Throwable? =
        when (value) {
            is Failure -> null
            else -> value as T
        }

    /**
     * Returns a shorthand for `getOrElse { null }` (see [getOrElse]).
     */
    @InlineOnly
    @SinceKotlin("1.3")
    public inline fun <R> getOrThrow(): R? =
        when {
            isFailure -> null
            else -> value as T
        }

    /**
     * Returns an instance that encapsulates the given [value] as successful value.
     */
    @Suppress("INAPPLICABLE_JVM_NAME")
    @InlineOnly
    @JvmName("getOrThrow")
    public
    }
R> Result<T>.getOrElse(onFailure: (exception: Throwable) -> R): R {
\n contract {\n     \ncallsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\n   }\n return when (val exception = exceptionOrNull()) {\n     null -> value as T\n     else -> onFailure(exception)\n   }\n
/*\n* An@InlineOnly\n@SinceKotlin("1.3")\n*/\npublic inline fun <R, T: R> Result<T>.getOrDefault(defaultValue: R): R {
\n contract {
     \ncallsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)\n   }\n return when (val exception = exceptionOrNull()) {\n     null -> value as T\n     else -> onFailure(exception)\n   }\n
/**\n* Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the\n* null * [defaultValue] if it is [failure][Result.isFailure].\n* * This function is a shorthand for `getOrElse { defaultValue }` (see [getOrElse]).\n*\n* An@InlineOnly\n@SinceKotlin("1.3")\n*/\npublic inline fun <R, T : R> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {\n    \n    val value = value // workaround for inline classes BE\n    \n    return when (val exception = exceptionOrNull()) {\n        null -> this\n        else -> runCatching {\n            \n            transform(exception)\n        }\n    }\n
/**\n* Performs the given [action] on the encapsulated [Throwable] exception if this instance represents\n* [failure][Result.isFailure].\n* 
* An@InlineOnly\n@SinceKotlin("1.3")\n*/\npublic inline fun <R, T : R> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {\n    \n    contract {\n        \ncallsInPlace(action, InvocationKind.AT_MOST_ONCE)\n    }\n    return when (val exception = exceptionOrNull()) {\n        null -> this\n        else -> action(it)\n    }\n}
encapsulated value if this instance represents [success][Result.isSuccess].

* Returns the original `Result` unchanged.

+/*@InlineOnly*/ Result.onSuccess(action: (value: T) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    if (isSuccess) action(value as T)
    return this
}

// -------------------
/*
 * Copyright 2010-2020 JetBrains
 * s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines

import kotlin.contracts.*
import kotlin.coroutines.intrinsics.*
import kotlin.internal.InlineOnly

/**
 * Interface representing a continuation after a suspension point that returns a value of type `T`.
 */
@SinceKotlin("1.3")
public interface Continuation<in T> {
    /**
     * The context of the coroutine that corresponds to this continuation.
     */
    public val context: CoroutineContext

    /**
     * Resumes the execution of the corresponding coroutine passing a successful or failed [result] as the return value of the last suspension point.
     */
    public fun resumeWith(result: Result<T>)
}

/**
 * Classes and interfaces marked with this annotation are restricted when used as receivers for extension `suspend` functions. These `suspend` extensions can only invoke other member or extension `suspend` functions on this particular `receiver` and are restricted from calling arbitrary suspension functions.
 */
@SinceKotlin("1.3")
@Target(AnnotationTarget.CLASS)
@Retention(AnnotationRetention.BINARY)
public annotation class RestrictsSuspension

/**
 * Resumes the execution of the corresponding coroutine passing [value] as the return value of the last suspension point.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation<T>.resume(value: T): Unit =
    resumeWith(Result.success(value))

/**
 * Resumes the execution of the corresponding coroutine so that the [exception] is re-thrown right after the last suspension point.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation<T>.resumeWithException(exception: Throwable): Unit =
    resumeWith(Result.failure(exception))

/**
 * Creates a `Continuation` instance with the given [context] and implementation of [resumeWith] method.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation(
    context: CoroutineContext,
    crossinline resumeWith: (Result<T>) -> Unit
): Continuation<T> =
    object : Continuation<T> {
        override val context: CoroutineContext
            get() = context

        override fun resumeWith(result: Result<T>) =
            resumeWith(result)
    }

/**
 * Creates a coroutine without a receiver and with result type [T].
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <T> (suspend () -> T).createCoroutine(
    completion: Continuation<T>
): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(completion).intercepted(), COROUTINE_SUSPENDED)

/**
 * Creates a coroutine with receiver type [R] and result type [T].
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <R, T> (suspend R.() -> T).createCoroutine(
    receiver: R,
    completion: Continuation<T>
): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(), COROUTINE_SUSPENDED)

/**
 * Starts a coroutine without a receiver and with result type [T].
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <T> (suspend () -> T).startCoroutine(
    completion: Continuation<T>
) {
    createCoroutineUnintercepted(completion).intercepted().resume(Unit)
}

/**
 * Starts a coroutine with receiver
*/
type [R] and result type [T].

* This function creates and starts a new, fresh instance of suspendable computation every time it is invoked.

* The [completion] continuation is invoked when the coroutine completes with a result or an exception.

* Obtains the current continuation instance inside suspend functions and suspends the currently running coroutine.

* In this function both [Completion.resume] and [Completion.resumeWithException] can be used either synchronously or asynchronously later in the same thread or from a different thread of execution. Subsequent invocation of any resume function will produce an [IllegalStateException].

* Starts an unintercepted coroutine without a receiver and with result type [T] and executes it until its first suspension.

* Returns the result of the coroutine or throws its exception if it does not suspend or if it suspends.

* In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.

* The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation context is established.

* This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.

* Starts an unintercepted coroutine without a receiver and with result type [T] and executes it until its first suspension.

* Returns the result of the coroutine or throws its exception if it does not suspend or if it suspends.

* In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.

* The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation context is established.

* This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.

* Starts an unintercepted coroutine with receiver type [R] and result type [T] and executes it until its first suspension.

* Returns the result of the coroutine or throws its exception if it does not suspend or if it suspends.

* In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.

* The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation context is established.

* This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of the suspended coroutine using a reference to the suspending function.
context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts `null` state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on the returned [Completion] instance may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

* [Completion.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

[Implementation.intercepted] that might be present in the completion's [CoroutineContext].

* It is the invoker's responsibility to ensure that a proper invocation context is established.

* Note that [completion] of this function may get invoked in an arbitrary context.

* [Completion.intercepted] can be used to acquire the intercepted continuation.

* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `null` both the coroutine and [completion] happens in the invocation context established by `null`.

* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.
actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")
actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")
public actual open class RuntimeException actual constructor(message: String?, cause: Throwable?) : Exception(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message)
{    actual constructor() : this(null)
}

public actual open class ConcurrentModificationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message)
{    actual constructor() : this(null)
}

public actual open class NullPointerException actual constructor(message: String?) : RuntimeException(message)
{    actual constructor() : this(null)
}

public actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message)
{    actual constructor() : this(null)
}

public actual open class AssertionError
@SinceKotlin("1.4")
constructor(message: String?, cause: Throwable?) : Error(message, cause)
{    actual constructor() : this(message.toString(), message as? Throwable)
}

public actual open class NoSuchElementException actual constructor(message: String?) : RuntimeException(message)
{    actual constructor() : this(null)
}

@SinceKotlin("1.3")
public actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message)
{    actual constructor() : this(null)
}

public actual open class NoWhenBranchMatchedException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class UninitializedPropertyAccessException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
{    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. Use of this source code is
 * governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.js

@kotlin.internal.InlineOnly
internal inline fun jsDeleteProperty(obj: Any, property: Any) {
    js("delete obj[property]")
}

@kotlin.internal.InlineOnly
internal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =
    js("lhs | rhs").unsafeCast<Int>()

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */
package kotlin.math

/**
 * Returns this value with the sign bit same as of the
 * [sign] value.
 * * Returns this value with the sign bit same as of the
 * [sign] value.
 * * Returns this value with the sign bit same as of the
 */
@SinceKotlin("1.2")
public actual
fun Double.withSign(sign: Double): Double {
    val thisSignBit = js("Kotlin").doubleSignBit(this).unsafeCast<Int>()
    val newSignBit = js("Kotlin").doubleSignBit(sign).unsafeCast<Int>()
    return if (thisSignBit == newSignBit) this else -this
}

---

The Open Source Used In DNAC Platform Arica (8097)
THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt

See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

//

import kotlin.js.*

/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 */
public actual fun CharSequence.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.collections

/*
 * THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
 * See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
 */

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.elementAt(index: Int): UInt {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun ULongArray.elementAt(index: Int): ULong {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.elementAt(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.elementAt(index: Int): UShort {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.asList(): List<UInt> {
    return object : AbstractList<UInt>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UInt): Boolean = this@asList.contains(element)
        override fun get(index: Int): UInt {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun ULongArray.asList(): List<ULong> {
    return object : AbstractList<ULong>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: ULong): Boolean = this@asList.contains(element)
        override fun get(index: Int): ULong {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
null

abstract fun checksElementIndex(index, size): return this.asList[index]

override fun indexof(element: UByte): Int = if ((element as Any?) !is UByte) return -1
return this.asList.indexOf(element)

override fun lastIndexOf(element: UByte): Int = if ((element as Any?) !is UByte) return -1
return this.asList.lastIndexOf(element)

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {

        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort = 
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        override fun indexOf(element: UShort): Int = if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        override fun lastIndexOf(element: UShort): Int = if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:Suppress("DEPRECATION_ERROR")
package jquery.ui

import jquery.JQuery
import kotlin.js.Json

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.buttonset(): JQuery = asDynamic().buttonset()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.dialog(): JQuery = asDynamic().dialog()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.dialog(params: Json): JQuery = asDynamic().dialog(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.dialog(mode: String): JQuery = asDynamic().dialog(mode)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.dialog(mode: String, param: String, value: Any?): JQuery = asDynamic().dialog(mode, param, value)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.button(): JQuery = asDynamic().button()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.accordion(): JQuery = asDynamic().accordion()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.draggable(params: Json): JQuery = asDynamic().draggable(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.selectable(): JQuery = asDynamic().selectable()

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin


public actual fun interface Comparator<T> {

    @JsName("compare")
    public actual fun compare(a: T, b: T): Int
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.js

import kotlin.annotation.AnnotationTarget.*

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic", level = DeprecationLevel.ERROR)
public annotation class nativeGetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic", level = DeprecationLevel.ERROR)
public annotation class nativeSetter

@Target(CLASS, FUNCTION, PROPERTY)
internal
annotation class library(public val name: String = `\"\"")

@Target(CLASS)
internal annotation class marker

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 * This may be useful in the following cases:
 * * There are two functions for which the compiler gives same name in JavaScript, you can mark one with @JsName(...) to prevent the compiler from reporting error.
 * * You are writing a JavaScript library in Kotlin. The compiler produces mangled names for functions with parameters, which is unnatural for usual Java script developer.
 * You can put @JsName(...) on functions you want to be available from JavaScript.
 * * For some reason you want to rename declaration, e.g. there’s common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 * * Example:

Example:

```
annotation class library(public val name: String = `\"\"")

@Target(CLASS)
internal annotation class marker

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 * This may be useful in the following cases:
 * * There are two functions for which the compiler gives same name in JavaScript, you can mark one with @JsName(...) to prevent the compiler from reporting error.
 * * You are writing a JavaScript library in Kotlin. The compiler produces mangled names for functions with parameters, which is unnatural for usual Java script developer.
 * You can put @JsName(...) on functions you want to be available from JavaScript.
 * * For some reason you want to rename declaration, e.g. there’s common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 * * Example:

Example:

```

```
applied to either files or top-level declarations.

It is currently prohibited to export the following kinds of declarations:

- `expect` declarations
- inline functions with reified type parameters
- `suspend` functions
- secondary constructors without `@JsName`
- extension properties
- enum classes
- `dynamic`, `Any`, `String`, `Boolean`, `Byte`, `Short`, `Int`, `Float`, `Double`
- `ByteArray`, `CharArray`, `IntArray`, `FloatArray`, `DoubleArray`
- `Array<exportable-type>`
- `Function` types with exportable parameters and return types
- `external` or `@JsExport` classes and interfaces
- `Nullable` counterparts of types above
- `Unit` type return. Must not be nullable

This annotation is experimental, meaning that restrictions mentioned above are subject to change.

```kotlin
@ExperimentalJsExport
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
@SinceKotlin("1.3")
public actual annotation class JsExport
```

---

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.jvm

// these are used in common generated code in stdlib

// TODO: find how to deprecate these ones

@Target( AnnotationTarget.FIELD)
@Retention( AnnotationRetention.SOURCE)
public actual annotation class Volatile

@Target( AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)
@Retention( AnnotationRetention.SOURCE)
public actual annotation class Synchronized

```
```
maps that are built (frozen) must throw `UnsupportedOperationException`.

```kotlin
    internal open fun checkIsMutable(): Unit {}
```

```kotlin
    override fun equals(other: Any?): Boolean {
        if (other === this) return true
        if (other !is Set<*>) return false
        return AbstractSet.setEquals(this, other)
    }
```

```kotlin
    override fun hashCode(): Int = AbstractSet.unorderedHashCode(this)
```

```kotlin
    public actual open class ArrayList<E> internal constructor(private var array: Array<Any?>) : AbstractMutableList<E>(), MutableList<E>, RandomAccess {
        private var isReadOnly: Boolean = false

        /**
         * Creates an empty [ArrayList].
         */
        public actual constructor() : this(emptyArray()) {}

        /**
         * Creates an empty [ArrayList].
         * @param initialCapacity initial capacity (ignored)
         */
        @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
        public actual constructor(initialCapacity: Int = 0) : this(emptyArray()) {}

        /**
         * Creates an [ArrayList] filled from the [elements] collection.
         */
        public actual constructor(elements: Collection<E>) :
```

```kotlin
        actual override val size: Int get() = array.size
```

```kotlin
        actual override fun get(index: Int): E = array[rangeCheck(index)] as E
```

```kotlin
        actual override fun set(index: Int, element: E): E {
            checkIsMutable()
            rangeCheck(index)
            @Suppress("UNCHECKED_CAST")
            return array[index].apply { array[index] = element } as E
        }
```

```kotlin
        actual override fun add(element: E): Boolean {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        override fun removeAt(index: Int): E {
            checkIsMutable()
            rangeCheck(index)
            modCount++
            return if (index == lastIndex)
                array.asDynamic().pop()
            else
                array.asDynamic().splice(index, 1)
        }
```

```kotlin
        override fun remove(element: E): Boolean {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            modCount++
            array.asDynamic().splice(fromIndex, toIndex - fromIndex)
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
            }
            modCount++
            return true
        }
```

```kotlin
        actual override fun removeRange(fromIndex: Int, toIndex: Int) {
            checkIsMutable()
            for (index in array.indices) {
                if (array[index] == element) {
                    array.asDynamic().splice(index, 1)
                    modCount++
                    return true
                }
            }
            return false
        }
```

```kotlin
        actual override fun addAll(elements: Collection<E>): Boolean {
            if (elements.isEmpty()) return false
            when (index) {
                size -> return addAll(elements)
                0 -> array = elements.toTypedArray<Any?>() + array
                else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
```
clear() {
    checkIsMutable()
    array = emptyArray()
    modCount++
}

fun actual override fun indexOf(element: E): Int = array.indexOf(element)

fun actual override fun lastIndexOf(element: E): Int = array.lastIndexOf(element)

override fun toString() = arrayToString(array)
override fun toArray(): Array<Any?> = js("[]").slice.call(array)

internal override fun checkIsMutable() {
    if (isReadOnly) throw UnsupportedOperationException()
}

private fun rangeCheck(index: Int) = index.apply {
    AbstractList.checkElementIndex(index, size)
}

private fun insertionRangeCheck(index: Int) = index.apply {
    AbstractList.checkPositionIndex(index, size)
}

internal fun <T> sortArrayWith(array: Array<out T>, comparator: Comparator<in T>) {
    if (getStableSortingIsSupported()) {
        array.asDynamic().sort(comparator)
    } else {
        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, comparator)
    }
}

internal fun <T> sortArrayWith(array: Array<out T>, comparison: (T, T) -> Int) {
    if (getStableSortingIsSupported()) {
        array.asDynamic().sort(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, Comparator(comparison))
    }
}

internal fun <T> sortArrayWith(array: Array<out T>, fromIndex: Int, toIndex: Int, comparator: Comparator<in T>) {
    if (fromIndex < toIndex - 1) {
        mergeSort(array.unsafeCast<Array<T>>(), fromIndex, toIndex - 1, comparator)
    }
}

internal fun <T : Comparable<T>> sortArray(array: Array<out T>) {
    if (getStableSortingIsSupported()) {
        val comparison = { a: T, b: T -> a.compareTo(b) }
        array.asDynamic().sort(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, naturalOrder())
    }
}

internal fun <T> mergeSort(array: Array<T>, start: Int, endInclusive: Int, comparator: Comparator<in T>): Array<T> {
    val buffer = arrayOfNulls<Any?>(array.size).unsafeCast<Array<T>>()
    val target = if (left === buffer) array else buffer
    // Merge.
    var leftIndex = start
    var rightIndex = endInclusive
    for (i in start..endInclusive) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                target[i] = leftValue
                leftIndex++
            }
            leftIndex <= median -> {
                target[i] = leftValue
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = rightValue
                rightIndex++
                Unit  // TODO: Fix KT-31506
            }
        }
    }
    return target
}

internal fun <T> mergeSort(array: Array<T>, buffer: Array<T>, start: Int, end: Int, comparator: Comparator<in T>): Array<T> {
    if (start == end) {
        return array
    }

    val median = (start + end) / 2
    val left = mergeSort(array, buffer, start, median, comparator)
    val right = mergeSort(array, buffer, median + 1, end, comparator)

    val target = if (left === buffer) array else buffer
    // Merge.
    var leftIndex = start
    var rightIndex = median + 1
    for (i in start..end) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                target[i] = leftValue
                leftIndex++
            }
            leftIndex <= median -> {
                target[i] = leftValue
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = rightValue
                rightIndex++
                Unit  // TODO: Fix KT-31506
            }
        }
    }
    return target
}

private fun <T> getStableSortingIsSupported(): Boolean {
    val array = js("[]").unsafeCast<Array<Int>()
    // known implementations may use stable sort for arrays of up to 512 elements
    // so we create slightly more elements to test stability
    for (index in 0 until 600) array.asDynamic().push(index)
    val comparison = { a: Int, b: Int -> (a and 3) - (b and 3) }
    array.asDynamic().sort(comparison)
    for (index in 1 until array.size) {
        val a = array[index - 1]
        val b = array[index]
        if ((a and 3) == (b and 3) && a >= b) return false
    }
    _stableSortingIsSupported = true
    return true
}

    if (start == end) {
        return array
    }

    val median = (start + end) / 2
    val left = mergeSort(array, buffer, start, median, comparator)
    val right = mergeSort(array, buffer, median + 1, end, comparator)

    val target = if (left === buffer) array else buffer
    // Merge.
    var leftIndex = start
    var rightIndex = median + 1
    for (i in start..end) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                target[i] = leftValue
                leftIndex++
            }
            leftIndex <= median -> {
                target[i] = leftValue
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = rightValue
                rightIndex++
                Unit  // TODO: Fix KT-31506
            }
        }
    }
    return target
}

    if (start == end) {
        return array
    }

    val median = (start + end) / 2
    val left = mergeSort(array, buffer, start, median, comparator)
    val right = mergeSort(array, buffer, median + 1, end, comparator)

    val target = if (left === buffer) array else buffer
    // Merge.
    var leftIndex = start
    var rightIndex = median + 1
    for (i in start..end) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                target[i] = leftValue
                leftIndex++
            }
            leftIndex <= median -> {
                target[i] = leftValue
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = rightValue
                rightIndex++
                Unit  // TODO: Fix KT-31506
            }
        }
    }
    return target
}
element is UByteArray -> element.contentHashCode()
  element is UShortArray -> element.contentHashCode()
  element is UIntArray -> element.contentHashCode()
  element is ULongArray -> element.contentHashCode()

  else                    -> element.hashCode()
} result = 31 * result + elementHash
    }
    return result
} /*
 * Copyright 2010-2018 JetBrains s.r.o. and
 * Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * */

package kotlin.collections

internal interface EqualityComparator {
    /**
     * Subclasses must override to return a value indicating
     * whether or not two keys or values are equal:
     * ```
     * abstract fun equals(value1: Any?, value2: Any?): Boolean
     * ```
     * ```
     * Subclasses must override to return the hash code of a given key:
     * ```
     * ```
     * abstract fun getHashCode(value: Any?): Int
     * ```
     * ```
     */
    abstract fun equals(value1: Any?, value2: Any?): Boolean

    /**
     * Subclasses must override to return the hash code of a given key:
     * ```
     * ```
     * abstract fun getHashCode(value: Any?): Int
     * ```
     */
    abstract fun getHashCode(value: Any?): Int

    object HashCode : EqualityComparator {
        override fun equals(value1: Any?, value2: Any?): Boolean = value1 == value2

        override fun getHashCode(value: Any?): Int = value?.hashCode() ?: 0
    }
} /*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * */
/*
 * Based on GWT AbstractHashMap
 * Copyright 2008 Google Inc.
 */

package kotlin.collections

import kotlin.collections.MutableMap.MutableEntry

/**
 * Hash table based implementation of the 
 * [MutableMap] interface.
 *
 * This implementation makes no guarantees regarding the order of enumeration of 
 * [keys], [values] and [entries] collections.
 * ```
 * Classes that extend HashMap and implement 'build()' (freezing) operation
 * have to make sure to mutating methods check 'checkIsMutable' method public abstract class HashMap<K, V> :
 * AbstractMutableMap<K, V>, MutableMap<K, V> |\n * private inner class EntrySet :
 * AbstractMutableSet<MutableEntry<K, V>> |\n * override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on entries") |\n * override fun clear() {
 * this@HashMap.clear() |\n * override operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element) |\n * override fun remove(element: MutableEntry<K, V>): Boolean = removeEntry(element) |\n * override val size: Int get() = this@HashMap.size |\n * }
 */

public actual open class HashMap<K, V> : AbstractMutableMap<K, V>, MutableMap<K, V> {

    private inner class EntrySet :
        AbstractMutableSet<MutableEntry<K, V>>() {

            override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on entries")

            override fun clear() {
                this@HashMap.clear()
            }

            override operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)

            override fun remove(element: MutableEntry<K, V>): Boolean = removeEntry(element)

            override val size: Int get() = this@HashMap.size
        }

    /**
     * Internal implementation of the map: either string-based or hashcode-based.
     * ```
     */
    private val internalMap: InternalMap<K, V>

    private val equality: EqualityComparator

    internal constructor(internalMap: InternalMap<K, V>) : super() {
        this.internalMap = internalMap
        this.equality = equality
    }

    constructor() : this(InternalHashCodeMap(EqualityComparator.HashCode))

    constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : this(initialCapacity, loadFactor)

    constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

    constructor(original: Map<out K, V>) : this() {
        putAll(original)
    }

    override fun clear() {
        internalMap.clear()
        // structureChanged(this)
    }

    override fun containsKey(key: K): Boolean = internalMap.containsKey(key)

    override fun containsValue(value: V): Boolean = internalMap.containsValue(value)

    override fun get(key: K): V? = internalMap.get(key)

    override fun put(key: K, value: V): V? = internalMap.put(key, value)

    override fun remove(key: K): V? = internalMap.remove(key)

    protected open fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()
override val size: Int get() = internalMap.size

/**
 * Constructs the specialized implementation of HashMap with [String] keys, which stores the keys as properties of JS object without hashing them.
 */
@public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> { return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs) } }

/**
 * Constructs a new [HashMap] filled with the elements of the specified collection.
 */
public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
    return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs) }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */
/*
 * Based on GWT InternalHashCodeMap
 * Copyright 2008 Google Inc.
 */

package kotlin.collections

/**
 * A simple wrapper around JavaScriptObject to provide java.util.Map-like semantics for any key type.
 * A key's hashCode is the index in backingMap which should contain that key. Since several keys may have the same hash, each value in hashCodeMap is actually an array containing all entries whose keys share the same hash.
 * A simple wrapper around JavaScriptObject to provide [java.util.Map]-like semantics for any key type.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) {
    this(initialCapacity, 0.0f)
}

actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

internal constructor(map: HashMap<E, Any>) {
    this.map = map
}

actual override fun add(element: E): Boolean {
    val old = map.put(element, this)
    return old == null
}

actual override function override fun clear() {
    map.clear()
}

actual override fun contains(element: E): Boolean = map.containsKey(element)

actual override fun isEmpty(): Boolean = map.isEmpty()

actual override fun iterator(): Collection<E> = map.keys.iterator()

actual override fun remove(element: E): Boolean = map.remove(element) != null

actual override val size: Int get() = map.size

/**
 * Creates a new instance of the specialized implementation of [HashSet] with the specified [String] elements, which elements the keys as properties of JS object without hashing them.
 */
public fun stringSetOf(vararg elements: String): HashSet<String> {
    return HashSet(stringMapOf<Any>()).apply { addAll(elements) }
}
A simple wrapper around JavaScript Map for key type is string. Though this map is instantiated only with K=String, the K type is not fixed to String statically, because we want to have it erased to Any in order not to generate type-safe override bridges for [get], [contains], [remove] etc, if they ever are generated.

```kotlin
package kotlin.collections

import kotlin.collections.MutableMap.MutableEntry

/**
 * A simple wrapper around JavaScript Map for key type is string.
 *
 * Though this map is instantiated only with K=String, the K type is not fixed to String statically, because we want to have it erased to Any in order not to generate type-safe override bridges for [get], [contains], [remove] etc, if they ever are generated.
 */
internal class InternalStringMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {

    private var backingMap: dynamic = createJsMap()
    override var size: Int = 0
        private set

    override operator fun contains(key: K): Boolean {
        if (key !is String) return false
        return backingMap[key] !== undefined
    }

    override operator fun get(key: K): V? {
        if (key !is String) return null
        val value = backingMap[key]
        return if (value !== undefined) value.unsafeCast<V>() else null
    }

    override fun put(key: K, value: V): V? {
        require(key is String)
        val oldValue = backingMap[key]
        backingMap[key] = value
        if (oldValue === undefined) {
            size++
            return null
        } else {
            return oldValue.unsafeCast<V>()
        }
    }

    override fun remove(key: K): V? {
        if (key !is String) return null
        val value = backingMap[key]
        if (value !== undefined) {
            jsDeleteProperty(backingMap, key)
            size--
            return value.unsafeCast<V>()
        } else {
            return null
        }
    }

    override fun clear() {
        backingMap = createJsMap()
        size = 0
    }

    override fun iterator(): MutableIterator<MutableEntry<K, V>> {
        return object : MutableIterator<MutableEntry<K, V>> {
            private val keys: Array<String> = js("Object").keys(backingMap)
            private val iterator = keys.iterator()
            private var lastKey: String? = null

            override fun hasNext(): Boolean = iterator.hasNext()

            override fun next(): MutableEntry<K, V> {
                val key = iterator.next()
                lastKey = key
                @Suppress("UNCHECKED_CAST")
                return newMapEntry(key as K)
            }

            override fun remove() {
                @Suppress("UNCHECKED_CAST")
                this@InternalStringMap.remove(checkNotNull(lastKey) as K)
            }
        }
    }

    private fun newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {
        override val key: K get() = key
        override val value: V get() = this@InternalStringMap[key].unsafeCast<V>()

        override fun setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()

        override fun get(): V = this@InternalStringMap[key].unsafeCast<V>()

        override fun hashCode(): Int = AbstractMap.entryHashCode(this)
        override fun equals(other: Any?): Boolean = AbstractMap.entryEquals(this, other)
        override fun toString(): String = AbstractMap.entryToString(this)

        @Suppress("UNCHECKED_CAST")
        fun @Suppress("UNCHECKED_CAST")
        return newMapEntry(key as K)
    }

    override fun override fun remove(entry: InternalStringMap.Entry<K, V>) {
        if (entry.key is String) {
            jsDeleteProperty(backingMap, entry.key)
            size--
        } else {
            return null
        }
    }
}
```

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
/*
 * Based on GWT LinkedHashMap
 * Copyright 2008 Google Inc.
 */
package kotlin.collections

import kotlin.collections.MutableMap.MutableEntry

/**
 * Hash table based implementation of the [MutableMap] interface, which additionally preserves the insertion order of entries during the iteration.
 */
public actual open class LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {

    /**
     * The entry we use includes next/prev pointers for a doubly-linked circular list with a head node. This reduces the special cases we have to deal with in the list operations.
     * Note that we duplicate the key from the underlying hash map so we can find the eldest entry. The alternative would have been to modify HashMap so more
     * of the codes was directly usable here, but this would have added some overhead to HashMap, or to reimplement most of the HashMap code here within small modifications. Paying a small storage cost only if you use Internal var
     */
    private class ChainEntry<K, V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V>(key, value) {

        internal var
```
```kotlin
isReadOnly = true

return this

actual override fun clear() {
    checkIsMutable()

    head = null

    override fun clone(): Any {
        return LinkedHashSet(this)
    }

    actual override fun clear() {
        checkIsMutable()

        head = null

        map.clear()

    }

    actual override fun containsKey(key: K): Boolean = map.containsKey(key)

    actual override fun containsValue(value: V): Boolean {
        var node: ChainEntry<K, V> = head ?: return false

        if (node.value == value) {
            return true
        }

        node = node.next!!

        while (node != head)

            return false

        override fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()

    }

    actual override operator fun get(key: K): V? = map.get(key)!!.value

    actual override fun put(key: K, value: V): V? {
        checkIsMutable()

        val old = map.get(key)

        if (old == null) {
            val newEntry = ChainEntry(key, value)
            map.put(key, newEntry)
            newEntry.addToEnd()

            return null
        } else {
            return old.setValue(value)
        }

    }

    actual override fun remove(key: K): V? {
        checkIsMutable()

        val entry = map.remove(key)

        if (entry != null) {
            entry.remove()
            return entry.value
        }

        return null
    }

    actual override val size: Int get() = map.size

    internal override fun checkIsMutable() {
        if (isReadOnly) throw UnsupportedOperationException()
    }
}

/**
 * Constructs the specialized implementation of
 */
public fun <K, V> linkedHashMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> {
    return LinkedHashMap(mapOf(Any() = Any())).apply {
        putAll(pairs)
    }
}

/**
 * The implementation of the 
 * interface, backed by a 
 */
public actual open class LinkedHashMap<E> : HashSet<E>, MutableSet<E> {

    internal constructor(map: LinkedHashMap<E, Any>) : super(map)

    /**
     * Constructs a new empty 
     */
    actual constructor() : super(LinkedHashMap<E, Any>())

    /**
     * Constructs a new 
     * filled with the elements of the specified collection.
     */
    actual constructor(elements: Collection<E>) : super(LinkedHashMap<E, Any>()) {
        addAll(elements)
    }

    /**
     * Creates a new instance of the specialized implementation of
     */
    public fun synchronized(lock: Any, block: () -> R): R {
        contract {
            callsInPlace(block, InvocationKind.EXACTLY_ONCE)
        }

        return block()
    }

    @PublishedApi
    internal fun build(): Set<E> {
        (map as LinkedHashMap).build()
        return this
    }

    override fun println() {
        print("\n")
    }

    open fun println(message: Any?) {
        print(message)
        println()
    }

    abstract fun print(message: Any?)

    open fun flush() {}
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("NodeJsOutput")
internal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {

    override fun println(message: Any?) {
        // TODO: Using local variable because of bug in block decomposition
    }
}
```

---

Open Source Used In DNAC Platform Arica  8111
lowering in IR backend

```kotlin
val messageString = String(message)
outputStream.write(messageString)
```

```kotlin
@JsName("OutputToConsoleLog")
internal class OutputToConsoleLog : BaseOutput() {
    override fun print(message: Any?) {
        console.log(message)
    }

    override fun println(message: Any?) {
        console.log(message)
    }
}
```

```kotlin
/** JsName used to make the declaration available outside of module to test it */
@JsName("OutputToConsoleLog")
internal class OutputToConsoleLog : BaseOutput() {
    override fun print(message: Any?) {
        console.log(message)
    }

    override fun println(message: Any?) {
        console.log(message)
    }
}
```

```kotlin
/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("BufferedOutput")
internal open class BufferedOutput : BaseOutput() {
    var buffer = ""

    override fun print(message: Any?) {
        buffer += String(message)
    }

    override fun flush() {
        buffer = ""
    }
}
```

```kotlin
/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {
    override fun override fun print(message: Any?) {
        var s = String(message)
        val i = s.nativeLastIndexOf("\n", 0)
        if (i >= 0) {
            buffer += s.substring(0, i)
            flush()
            s = s.substring(i + 1)
        }
        buffer += s
    }

    override fun flush() {
        console.log(buffer)
        buffer = ""
    }
}
```

```kotlin
/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("output")
internal var output = run {
    val isNode: Boolean = js("typeof process !== 'undefined' && process.versions && !!process.versions.node")
    if (isNode) NodeJsOutput(js("process.stdout")) else BufferedOutputToConsoleLog()
}
```

```kotlin
/** Prints the line separator to the standard output stream. */
public actual fun println() {
    output.println()
}
```

```kotlin
/** Prints the given [message] and the line separator to the standard output stream. */
public actual fun println(message: Any?) {
    output.println(message)
}
```

```kotlin
/** Prints the given [message] to the standard output stream. */
public actual fun print(message: Any?) {
    output.print(message)
}
```

```
Open Source Used In DNAC Platform Arica 8112
```
EmptyContinuation = Continuation<Any?><EmptyCoroutineContext> { result ->
    result.getOrThrow() } /*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/**
 * Exposes the [Date API](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public external class Date() {
    public constructor(milliseconds: Number)
    public constructor(dateString: String)
    public constructor(year: Int, month: Int)
    public constructor(year: Int, month: Int, day: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number)

    public fun getDate(): Int
    public fun getDay(): Int
    public fun getFullYear(): Int
    public fun getHours(): Int
    public fun getMilliseconds(): Int
    public fun getMinutes(): Int
    public fun getMonth(): Int
    public fun getSeconds(): Int
    public fun getTime(): Double
    public fun getTimezoneOffset(): Int
    public fun getUTCDate(): Int
    public fun getUTCDay(): Int
    public fun getUTCFullYear(): Int
    public fun getUTCMilliseconds(): Int
    public fun getUTCMonth(): Int
    public fun getUTCSeconds(): Int
    public fun toDateString(): String
    public fun toISOString(): String
    public fun toJSON(): Json
    public fun toLocaleDateString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleTimeString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toTimeString(): String
    public fun toUTCString(): String

    public companion object {
        public fun now(): Double
        public fun parse(dateString: String): Double
        public fun UTC(year: Int, month: Int): Double
        public fun UTC(year: Int, month: Int, day: Int): Double
        public fun UTC(year: Int, month: Int, day: Int, hour: Int): Double
        public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int): Double
    }
}

public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
    val result = js("new Object()").unsafeCast<Date.LocaleOptions>()
    init(result)
    return result
}

package kotlin.js

import org.w3c.dom.Document
import org.w3c.dom.Element
import kotlinx.dom.appendElement as newAppendElement
import kotlinx.dom.createElement as newCreateElement

/**
 * Creates a new element with the specified [name].
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.createElement' instead.",
    replaceWith = ReplaceWith("this.createElement(name, init)", "kotlinx.dom.createElement")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun Document.createElement(name: String, noinline init: Element.() -> Unit): Element = this.createElement(name, init)
initialized with the specified [init] function.

```kotlin
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.appendElement' instead.",
    replaceWith = ReplaceWith("this.appendElement(name, init)"),
    kotlin.dom.appendElement("\n")
)

/** Returns true if the element has the given CSS class style in its 'class' attribute
   */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.hasClass' instead.",
    replaceWith = ReplaceWith("this.hasClass(cssClass)"),
    kotlin.dom.hasClass("\n")
)

/**
 * Adds CSS class to element. Has no effect if all specified classes are already in class attribute of the element.
 * @return true if at least one class has been added
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.addClass' instead.",
    replaceWith = ReplaceWith("this.addClass(cssClasses)"),
    kotlin.dom.addClass("\n")
)

/**
 * Removes all [cssClasses] from element. Has no effect if all specified classes are missing in class attribute of the element.
 * @return true if at least one class has been removed
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.removeClass' instead.",
    replaceWith = ReplaceWith("this.removeClass(cssClasses)"),
    kotlin.dom.removeClass("\n")
)
```

---

**Open Source Used In DNAC Platform Arica 8114**
---
Open Source Used In DNAC Platform Arica 8116

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.collections

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each element to get its key.
 * A [Grouping] structure serves as an intermediate step in group-and-fold operations:
 * they group elements by their keys and then fold each group with some aggregating operation.
 * It is created by attaching a `keySelector: (T) -> K` function to a source of elements.
 * To get an instance of [Grouping] use one of 'groupingBy' extension functions:
 * - [Iterable.groupingBy]
 * - [Sequence.groupingBy]
 * - [Array.groupingBy]
 * - [CharSequence.groupingBy]
 * For the list of group-and-fold operations available, see the [extension functions](#extension-functions) for `Grouping`.
 */
@SinceKotlin("1.1")
public interface Grouping<T, out K> {
    /** Returns an [Iterator] over the elements of the source of this grouping. */
    fun sourceIterator(): Iterator<T>
    /** Extracts the key of an [element]. */
    fun keyOf(element: T): K
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,
 * passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation function is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 *  - `element`: the element from the source being aggregated;
 *  - `first`: indicates whether it's the first `element` encountered in the group.
 * @return a [Map] associating the key of each group with the result of aggregation of the group elements.
 * @sample samples.collections.Grouping.aggregateByRadix
 */
@SinceKotlin("1.1")
public inline fun <T, K, R> Grouping<T, K>.aggregate(
    operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R
): Map<K, R> {
    return aggregateTo(mutableMapOf<K, R>(), operation)
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially,
 * passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group;
 *  - `element`: the element from the source being aggregated;
 *  - `first`: indicates whether it's the first `element` encountered in the group.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 * @sample samples.collections.Grouping.aggregateByRadixTo
 */
@SinceKotlin("1.1")
public inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.aggregateTo(
    destination: M,
    operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R
): M {
    for (e in this.sourceIterator()) {
        val key = keyOf(e)
        val accumulator = destination[key]
        destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))
    }
    return destination
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is provided by [initialValueSelector] function.
 * @param initialValueSelector a function that provides an initial value of accumulator for each group.
 *  It's invoked with parameters:
 *  - `key`: the key of the group;
 *  - `element`: the first element being encountered in that group.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group;
 *  - `element`: the element from the source being accumulated.
 * @return a [Map] associating the key of each group with the result of accumulating the group elements.
 * @sample samples.collections.Grouping.foldByEvenLengthWithComputedInitialValue
 */
@SinceKotlin("1.1")
public inline fun <T, K, R> Grouping<T, K>.fold(
    initialValueSelector: (key: K, element: T) -> R,
    operation: (key: K, accumulator: R, element: T) -> R
): Map<K, R> =
    @Suppress("UNCHECKED_CAST")
    aggregate { key, acc, e, first -> operation(key, if (first)
initialValueSelector(key, e) else acc as R, e) }

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.
 * An initial value of accumulator is provided by [initialValueSelector] function.
 */

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is the same [initialValue] for each group.
 */

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.
 * An initial value of accumulator is the same [initialValue] for each group.
 */

/**
 * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group sequentially starting from the second element of the group, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is the first element of the group.
 */

/**
 * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group sequentially starting from the second element of the group, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is the key of the group this element belongs to.
 */

/**
 * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group sequentially starting from the second element of the group, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is the key of the group this element belongs to.
 */

/**
 * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group sequentially starting from the second element of the group, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.
 * An initial value of accumulator is the key of the group this element belongs to.
 */
@Suppress("UNCHECKED_CAST")
        if (first) e else operation(key, acc as S, e)
    }

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group in
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@sample samples.collections.Grouping.reduceByMaxVowelsTo

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments and stores the results in the given [destination] map.
 * An initial value of the accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * the current value of the accumulator from the source being folded,
 * the element from the source being folded,
public fun atan(value: Double): Double
    @Deprecated("Use kotlin.math.atan2 instead.", ReplaceWith("atan2(y, x)", "kotlin.math.atan2()"), level = DeprecationLevel.ERROR)
    public fun atan2(y: Double, x: Double): Double
    @Deprecated("Use kotlin.math.cos instead.", ReplaceWith("cos(value)", "kotlin.math.cos()"), level = DeprecationLevel.ERROR)
    public fun cos(value: Double): Double
    @Deprecated("Use kotlin.math.sin instead.", ReplaceWith("sin(value)", "kotlin.math.sin()"), level = DeprecationLevel.ERROR)
    public fun sin(value: Double): Double
    @Deprecated("Use kotlin.math.exp instead.", ReplaceWith("exp(value)", "kotlin.math.exp()"), level = DeprecationLevel.ERROR)
    public fun exp(value: Double): Double
    @Deprecated("Use kotlin.math.atan instead.", ReplaceWith("atan(value)", "kotlin.math.atan()"), level = DeprecationLevel.ERROR)
    public fun tan(value: Double): Double
    @Deprecated("Use kotlin.math.ln instead.", ReplaceWith("ln(value)", "kotlin.math.ln()"), level = DeprecationLevel.ERROR)
    public fun log(value: Double): Double
    @Deprecated("Use kotlin.math.pow instead.", ReplaceWith("base.pow(exp)", "kotlin.math.pow()"), level = DeprecationLevel.ERROR)
    public fun pow(base: Double, exp: Double): Double
    @Deprecated("Use kotlin.math.round instead.", ReplaceWith("round(value)", "kotlin.math.round()"), level = DeprecationLevel.ERROR)
    public fun round(value: Number): Int
    @Deprecated("Use kotlin.math.floor instead.", ReplaceWith("floor(value)", "kotlin.math.floor()"), level = DeprecationLevel.ERROR)
    public fun floor(value: Number): Int
    @Deprecated("Use kotlin.math.ceil instead.", ReplaceWith("ceil(value)", "kotlin.math.ceil()"), level = DeprecationLevel.ERROR)
    public fun ceil(value: Number): Int

    @PublishedApi
    internal fun trunc(value: Number): Double
    @PublishedApi
    internal fun sign(value: Number): Double

    @PublishedApi
    internal fun sinh(value: Double): Double
    @PublishedApi
    internal fun cosh(value: Double): Double
    @PublishedApi
    internal fun tanh(value: Double): Double
    @PublishedApi
    internal fun asinh(value: Double): Double
    @PublishedApi
    internal fun acosh(value: Double): Double
    @PublishedApi
    internal fun atanh(value: Double): Double
    @PublishedApi
    internal fun hypot(x: Double, y: Double): Double
    @PublishedApi
    internal fun expm1(value: Double): Double
    @PublishedApi
    internal fun log10(value: Double): Double
    @PublishedApi
    internal fun log2(value: Double): Double
    @PublishedApi
    internal fun log1p(value: Double): Double

    @PublishedApi
    internal fun clz32(value: Int): Int
}

/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use minOf or kotlin.math.min instead", ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use maxOf or kotlin.math.max instead", ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun max(a: Long, b: Long): Long = if (a >= b) a else b
j["prop"] + j.get(p) + 
     * ```
     *
     * will be generated:
     * ```js
     * function test(j, p) {
     *     return
     *         j["prop"] + j[p];
     * }]
     */
    operator fun get(propertyName: String): Any?

    /**
     * Calls
     * E.g. for the following code:
     * ```kotlin
     * fun test(j: Json, p: String, newValue: Any) {
     *     j["prop"] = 1
     *     j[p] = newValue
     * }
     * ```
     */
    operator fun set(propertyName: String, value: Any?): Unit

    /**
     * Returns a simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.
     */
    @public fun json(vararg pairs: Pair<String, Any?>): Json {
        val res: dynamic = js("({})")
        for ((name, value) in pairs) {
            res[name] = value
        }
        return res
    }

    /**
     * Adds key-value pairs from [other] to [this].
     * Returns the original receiver.
     */
    @public fun Json.add(other: Json): Json {
        val keys: Array<String> = js("Object").keys(other)
        for (key in keys) {
            if (other.asDynamic().hasOwnProperty(key)) {
                this[key] = other[key]
            }
        }
        return this
    }

    /**
     */
    @Suppress("NOT_DOCUMENTED")
    public external object JSON {
        public fun stringify(o: Any?): String
        public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?): String
        public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?: Any?): String
        public fun stringify(o: Any?, replacer: Array<String>?): String
        public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
        public fun stringify(o: Any?, replacer: Array<String>, space: String): String
        public fun <T> parse(text: String): T
        public fun <T> parse(text: String, reviver: ((key: String, value: Any?) -> Any?)?): T
    }

/**
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:Suppress("DEPRECATION_ERROR")
package kotlin.math

// region ================ Double Math
========================================

/** Computes the sine of the angle [x] given in radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Sine of \$x is \$(sin(x))")
 * }
 * fun test(x: Double) {
 *     println("Cosine of \$x is \$(cos(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.sin(x);
 * }]
 * ```
 */
@.SinceKotlin("1.2")
public actual inline fun sin(x: Double): Double =
    nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Cosine of \$x is \$(cos(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.cos(x);
 * }]
 * ```
 */
@.SinceKotlin("1.2")
public actual inline fun cos(x: Double):
    Double =
    nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Tangent of \$x is \$(tan(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.tan(x);
 * }]
 * ```
 */
@.SinceKotlin("1.2")
public actual inline fun tan(x: Double):
    Double =
    nativeMath.tan(x)

/** Computes the arc sine of the value [x] given in the range from `-PI/2` to `PI/2` radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Arc sine of \$x is \$(asin(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.asin(x);
 * }]
 * ```
 */
@SinceKotlin("1.2")
public actual inline fun asin(x: Double):
    Double =
    nativeMath.asin(x)

/** Computes the arc cosine of the value [x] given in the range from `[0.0, PI)` radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Arc cosine of \$x is \$(acos(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.acos(x);
 * }]
 * ```
 */
@SinceKotlin("1.2")
public actual inline fun acos(x: Double):
    Double =
    nativeMath.acos(x)

/** Computes the arc tangent of the value [x] given in the range from `-PI` to `[PI)` radians.
 * E.g. for the following code:
 * ```kotlin
 * fun test(x: Double) {
 *     println("Arc tangent of \$x is \$(atan(x))")
 * }
 * ```
 * will be generated:
 * ```js
 * function test(x) {
 *     return
 *         Math.atan(x);
 * }]
 * ```
 */
@SinceKotlin("1.2")
public actual inline fun atan(x: Double):
    Double =
    nativeMath.atan(x)
*@SinceKotlin("1.2")@InlineOnly
public actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x)

/**
 * Computes the hyperbolic sine of the value [x].
 *
 * Special cases:
 *   - `sinh(NaN)` is `NaN`
 *   - `sinh(±Inf)` is `±Inf`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun sinh(x: Double): Double = nativeMath.sinh(x)

/**
 * Computes the hyperbolic cosine of the value [x].
 *
 * Special cases:
 *   - `cosh(NaN)` is `NaN`
 *   - `cosh(±Inf)` is `±Inf`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun cosh(x: Double): Double = nativeMath.cosh(x)

/**
 * Computes the hyperbolic tangent of the value [x].
 *
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(±Inf)` is `±1.0`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)

/**
 * Computes the inverse hyperbolic sine of the value [x].
 *
 * Special cases:
 *   - `asinh(NaN)` is `NaN`
 *   - `asinh(±Inf)` is `±Inf`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun asinh(x: Double): Double = nativeMath.asinh(x)

/**
 * Computes the inverse hyperbolic cosine of the value [x].
 *
 * Special cases:
 *   - `acosh(NaN)` is `NaN`
 *   - `acosh(x)` is `NaN` when `x < 1`
 *   - `acosh(±Inf)` is `±Inf`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun acosh(x: Double): Double = nativeMath.acosh(x)

/**
 * Computes the inverse hyperbolic tangent of the value [x].
 *
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(x)` is `NaN` when `x > 1` or `x < -1`
 *   - `tanh(0.0)` is `±0.0`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun atanh(x: Double): Double = nativeMath.atanh(x)

/**
 * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.
 *
 * Special cases:
 *   - returns `+Inf` if any of arguments is infinite
 *   - returns `NaN` if any of arguments is `NaN`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun hypot(x: Double, y: Double): Double = nativeMath.hypot(x, y)

/**
 * Computes the square root of the value [x].
 *
 * Special cases:
 *   - `sqrt(x)` is `NaN` when `x < 0`
 *   - `sqrt(x)` is `+Inf` when `x` is infinite
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun sqrt(x: Double): Double = nativeMath.sqrt(x)

/**
 * Computes Euler's number `e` raised to the power of the value [x].
 *
 * Special cases:
 *   - `exp(NaN)` is `NaN`
 *   - `exp(+Inf)` is `+Inf`
 *   - `exp(-Inf)` is `0.0`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun exp(x: Double): Double = nativeMath.exp(x)

/**
 * Computes `exp(x) - 1`.
 *
 * Special cases:
 *   - `expm1(NaN)` is `NaN`
 *   - `expm1(+Inf)` is `+Inf`
 *   - `expm1(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun expm1(x: Double): Double = nativeMath.expm1(x)

/**
 * Computes the logarithm of the value [x] to the given [base].
 *
 * Special cases:
 *   - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`
 *   - `log(0.0, b)` is `-Inf`
 *   - `log(x, 1)` is `-Inf`
 *   - `log(x, ±Inf)` is `±Inf`
 */
@SinceKotlin("1.2")@InlineOnly
public fun log(x: Double, base: Double): Double {
    val ln = nativeMath.log(x)
    val lnBase = nativeMath.log(base)
    return ln / lnBase
}

/**
 * Computes `ln(x + 1)`.
 *
 * Special cases:
 *   - `ln1p(NaN)` is `NaN`
 *   - `ln1p(0.0)` is `-Inf`
 *   - `ln1p(x)` is `-Inf` for `x < -1`
 */
@SinceKotlin("1.2")@InlineOnly
public actual inline fun ln1p(x: Double): Double = nativeMath.expm1(x)
nativeMath.log1p(x)\n\n** Rounds the given value [x] to an integer towards positive infinity.\n\nSpecial cases:\n* -  `ceil(x)` is `\`x` where `x` is `NaN`,`+Inf` or `-Inf` or already a mathematical integer.\n* @return the value [x] having its fractional part truncated.\n* @return the value [x] to an integer towards zero.\n* @return the value [x] to an integer towards positive infinity.\n* @return the value [x] to an integer towards negative infinity.\n* Special cases:\n* -  `floor(x)` is `\`x` where `x` is `NaN`,`+Inf` or `-Inf` or already a mathematical integer.\n* @return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.\n* @return the largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.\n\n sign(x)\n\n** Returns the sign of this value:
* - `-1.0` if the value is negative,
* - zero if the value is zero,
* - `1.0` if the value is positive
\n Special case:
* - `NaN.sign` is `NaN`
\n absoluteValue\n\n** Returns the absolute value of this value:
* - `-1.0` if the value is negative,
* - zero if the value is zero,
* - `-1.0` if the value is positive
\n Special case:
* - `NaN.absoluteValue` is `NaN`
\n* @see abs function \n\n withSign(sign)\n\n** Returns the ulp (unit in the last place) of this value.
\n Special Cases:\n* - `NaN.ulp` is `NaN`
* - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`
* - `0.0.ulp` is `Double.MIN_VALUE`
* - `Double.MIN_VALUE` is `NaN`
* - `Double.MAX_VALUE` is `+Inf`
* - `Double.NEGATIVE_INFINITY` is `-Inf`
* - `Double.POSITIVE_INFINITY` is `+Inf`
* - `Double.NEGATIVE_INFINITY` is `-Inf`
* - `Double.MAX_VALUE` is `+Inf`
* - `Double.MIN_VALUE` is `-Inf`
* - `Double.MIN_VALUE` is `-Inf`
Returns the [Double] value nearest to this value in direction of negative infinity.

*Kotlin*

```kotlin
public actual fun Double.nextDown(): Double = when {
    this.isNaN() || this == Double.NEGATIVE_INFINITY -> this
    this == 0.0 -> -Double.MIN_VALUE
    else -> Double.fromBits(this.toRawBits() + if (this > 0) -1 else 1)  // Rounds this [Double] value to the nearest integer and converts the result to [Int] * Ties are rounded towards positive infinity.
}

/**
 * Returns the 

 * Special cases:
 *    - `x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`
 *    - `x.nextTowards(y) == x` when `x < Int.MIN_VALUE`
 *    - `x.nextTowards(y) == y` when `x > Int.MAX_VALUE`
 */
public actual fun Double.nextTowards(to: Double): Double = when {
    this.isNaN() || to.isNaN() -> Double.NaN
    to == this -> to  // Rounds this [Double] value to the nearest integer and converts the result to [Long]  * Ties are rounded towards positive infinity.
    to < this && to.nextUp() == this -> to
    else /* to > this */ -> to.nextDown()
}

/**
 * Rounds this 

 * Special cases:
 *    - `x.roundToInt() == Int.MAX_VALUE` when `x > Int.MAX_VALUE`
 *    - `x.roundToInt() == Int.MIN_VALUE` when `x < Int.MIN_VALUE`
 */
public actual fun Double.roundToInt(): Int = when {
    isNaN() -> throw IllegalArgumentException("Cannot round NaN value.")
    this > Int.MAX_VALUE -> Int.MAX_VALUE
    this < Int.MIN_VALUE -> Int.MIN_VALUE
    else ->
        nativeMath.round(this).unsafeCast<Double>().toInt()
}

/**
 * Rounds this 

 * Special cases:
 *    - `x.roundToLong() == Long.MAX_VALUE` when `x > Long.MAX_VALUE`
 *    - `x.roundToLong() == Long.MIN_VALUE` when `x < Long.MIN_VALUE`
 */
public actual fun Double.roundToLong(): Long = when {
    isNaN() -> throw IllegalArgumentException("Cannot round NaN value.")
    this > Long.MAX_VALUE -> Long.MAX_VALUE
    this < Long.MIN_VALUE -> Long.MIN_VALUE
    else ->
        nativeMath.round(this).unsafeCast<Double>().toLong()
}
```

// endregion


// region ================

Float Math ========================================

/** Computes the sine of the angle 

 * Special cases:
 *    - `sin(x)` is `NaN` when `abs(x) > 1`
 */
public actual inline fun sin(x: Float): Float =
    nativeMath.sin(x.toDouble()).toFloat()

/** Computes the cosine of the angle 

 * Special cases:
 *    - `cos(x)` is `NaN` when `abs(x) > 1`
 */
public actual inline fun cos(x: Float): Float =
    nativeMath.cos(x.toDouble()).toFloat()

/** Computes the tangent of the angle 

 * Special cases:
 *    - `tan(x)` is `NaN` when `abs(x) > 1`
 */
public actual inline fun tan(x: Float): Float =
    nativeMath.tan(x.toDouble()).toDouble().toFloat()  // Computes the arc tangent of the value [x] given in radians.

/**
 * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 */
public actual fun atan2(y: Float, x: Float): Float =
    nativeMath.atan2(y.toDouble(), x.toDouble()).toFloat()  // Computes the arc tangent of the value [x] in radians.

```
the value \([x]\).`ln` * Special cases:`ln` * - `sinh(NaN)` is `NaN`'`n` * `-sinh(Inf)` is `+Inf`'`n` * `-sinh(-Inf)` is `-Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `sinh(x: Float)`: Float = nativeMath.sinh(x.toDouble()).toFloat()`n`/* Computes the hyperbolic cosine of the value \([x]\).`ln` * Special cases:`ln` * - `cosh(NaN)` is `NaN`'`n` * `-cosh(Inf)` is `+Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `cosh(x: Float)`: Float = nativeMath.cosh(x.toDouble()).toFloat()`n`/* Computes the hyperbolic tangent of the value \([x]\).`ln` * Special cases:`ln` * - `tanh(NaN)` is `NaN`'`n` * `-tanh(Inf)` is `1.0`'`n` * `-tanh(-Inf)` is `-1.0`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `tanh(x: Float)`: Float = nativeMath.tanh(x.toDouble()).toFloat()`n`/* Computes the inverse hyperbolic sine of the value \([x]\).`ln` * Special cases:`ln` * - `asinh(NaN)` is `NaN`'`n` * `-asinh(Inf)` is `+Inf`'`n` * `-asinh(-Inf)` is `-Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `asinh(x: Float)`: Float = nativeMath.asinh(x.toDouble()).toFloat()`n`/* Computes the inverse hyperbolic cosine of the value \([x]\).`ln` * Special cases:`ln` * - `acosh(NaN)` is `NaN`'`n` * `-acosh(Inf)` is `+Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `acosh(x: Float)`: Float = nativeMath.acosh(x.toDouble()).toFloat()`n`/* Computes the inverse hyperbolic tangent of the value \([x]\).`ln` * Special cases:`ln` * - `tanhh(x)` is `NaN`'`n` * `-tanh(1.0)` is `+Inf`'`n` * `-tanh(-1.0)` is `-Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `atanh(x: Float)`: Float = nativeMath.atanh(x.toDouble()).toFloat()`n`/* Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.`ln` * Special cases:`ln` * - returns `+Inf` if any of arguments is infinity`n` * - returns `NaN` if any of arguments is `NaN` and the other is not infinite`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `hypot(x: Float, y: Float)`: Float = nativeMath.hypot(x.toDouble(), y.toDouble()).toFloat()`n`/* Computes `exp(x)` without intermediate overflow or underflow.`ln` * Special cases:`ln` * - `exp(NaN)` is `NaN`'`n` * `-exp(Inf)` is `+Inf`'`n` * `-exp(-Inf)` is `0.0`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `exp(x: Float)`: Float = nativeMath.exp(x.toDouble()).toFloat()`n`/* Computes `expm1(x)` without intermediate overflow or underflow.`ln` * Special cases:`ln` * - `expm1(NaN)` is `NaN`'`n` * `-expm1(Inf)` is `+Inf`'`n` * `-expm1(-Inf)` is `-1.0`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `expm1(x: Float)`: Float = nativeMath.expm1(x.toDouble()).toFloat()`n`/* Computes Euler's number `e` raised to the power of the value \([x]\).`ln` * Special cases:`ln` * - `exp(NaN)` is `NaN`'`n` * `-exp(Inf)` is `+Inf`'`n` * `-exp(-Inf)` is `0.0`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `exp(x: Float)`: Float = nativeMath.exp(x.toDouble()).toFloat()`n`/* Computes `exp(x)` - 1.`ln` * Special cases:`ln` * - see [exp] function.`ln` * @see [exp] function.`ln` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `log(x: Float)`: Float = nativeMath.log(x.toDouble()).toFloat()`n`/* Computes the logarithm of the value \([x]\) to the given base.`ln` * Special cases:`ln` * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`'`n` * `-log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0`'`n` * `-log(Inf, b)` is `+Inf` for `b > 1` and `-Inf` for `b < 1`'`n` * `-log(0.0, b)` is `-Inf` for `b > 1` and `+Inf` for `b < 1`'`n` * See also logarithm functions for common fixed bases: [In], [Log10] and [Log2].`ln` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `log(x: Float)`: Float = nativeMath.log(x.toDouble()).toFloat()`n`/* Computes the natural logarithm (base `E`) of the value \([x]\).`ln` * Special cases:`ln` * - `ln(NaN)` is `NaN`'`n` * `-ln(x)` is `NaN` when `x < 0.0`'`n` * `-ln(Inf)` is `+Inf`'`n` * `-ln(0.0)` is `-Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `ln(x: Float)`: Float = nativeMath.log(x.toDouble()).toFloat()`n`/* Computes the common logarithm (base 10) of the value \([x]\).`ln` * Special cases:`ln` * - see [In] function for special cases.`ln` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `log10(x: Float)`: Float = nativeMath.log10(x.toDouble()).toFloat()`n`/* Computes the binary logarithm (base 2) of the value \([x]\).`ln` * Special cases:`ln` * - see [In] function for special cases.`ln` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `log2(x: Float)`: Float = nativeMath.log2(x.toDouble()).toFloat()`n`/* Computes `lnp(NaN)` is `NaN`'`n` * `-lnp(x)` is `NaN` where `x < 0.0`'`n` * `-lnp(-1.0)` is `-Inf`'`n` * `-lnp(1.0)` is `+Inf`'`n` * `@SinceKotlin("1.2")`@InlineOnly`n` public actual inline function `lnp(x: Float)`: Float = nativeMath.lnp(x.toDouble()).toFloat()`n`/* Computes `lnp(NaN)` is `NaN`'`n` * `-lnp(x)` is `NaN` where `x < 0.0`'`n` * `-lnp(-1.0)` is `-Inf`'`n` * `-lnp(1.0)` is `+Inf`'`n` * see [In] function`n`
@see [expm1] function

*\@fn\n
\*\@since Kotlin("1.2")\n
**fun ln1p(x: Float): Float =\n**

nativeMath.log1p(x.toDouble()).toFloat()

/**
 * Rounds the given value [x] to an integer towards positive infinity.
 * @return the smallest Float value that is greater than or equal to the given value [x] and is a mathematically integer.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun ln(x: Float): Float = nativeMath.ceil(x.toDouble()).toFloat()

/**
 * Rounds the given value [x] to an integer towards zero.
 * @return the value [x] having its fractional part truncated.
 * Special cases:\n * - `ceil(x)` is \(x\) where \(x\) is \(\text{NaN}\) or +Inf or -Inf or already a mathematical integer.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun floor(x: Float): Float = nativeMath.floor(x.toDouble()).toFloat()

/**
 * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
 * @return the value [x] having its fractional part truncated.
 * Special cases:\n * - `round(x)` is \(x\) where \(x\) is \(\text{NaN}\) or +Inf or -Inf or already a mathematical integer.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun truncate(x: Float): Float = truncate(x.toDouble()).toFloat()

/**
 * Rounds the given value [x] towards the closest integer with ties rounded towards negative infinity.
 * @return the largest Float value that is smaller than or equal to the given value [x] and is a mathematical integer.
 * Special cases:\n * - `floor(x)` is \(x\) where \(x\) is \(\text{NaN}\) or +Inf or -Inf or already a mathematical integer.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun ceil(x: Float): Float = nativeMath.ceil(x.toDouble()).toFloat()

/**
 * Returns the absolute value of the given value [x].
 * @return the absolute value of the given value [x] and is a mathematical integer.
 * Special cases:\n * - `abs(NaN)` is \(\text{NaN}\)\n * - `abs(\pm Inf)` is \(\pm Inf\)
 * - `abs(0)` is \(0\)
 * - `abs(x)` is \(\text{NaN}\) where \(x\) is not a mathematical integer.
 * @see absoluteValue extension property for [Float]
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun abs(x: Float): Float = nativeMath.abs(x.toDouble()).toFloat()

/**
 * Returns the sign of the given value [x]:
 * -1.0 if the value is negative,
 * zero if the value is zero,
 * 1.0 if the value is positive
 * Special case:
 * `sign(NaN)` is `NaN`
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun sign(x: Float): Float = nativeMath.sign(x.toDouble()).toFloat()

/**
 * Returns the smaller of two values.
 * If either value is `NaN`, then the result is `NaN`.
 * Special cases:\n * - `min(\pm Inf, \pm Inf)` is `\pm Inf`
 * - `min(\pm Inf, x)` is `x` where `x` is not `NaN`
 * - `min(x, \pm Inf)` is `\pm Inf`
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun min(a: Float, b: Float): Float = nativeMath.min(a, b)

/**
 * Returns the greater of two values.
 * If either value is `NaN`, then the result is `NaN`.
 * Special cases:\n * - `max(\pm Inf, \pm Inf)` is `\pm Inf`
 * - `max(\pm Inf, x)` is `x` where `x` is not `NaN`
 * - `max(x, \pm Inf)` is `\pm Inf`
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun max(a: Float, b: Float): Float = nativeMath.max(a, b)

/**
 * Returns the absolute value of this value.
 * Special cases:
 * - `abs(NaN)` is `NaN`
 * @see abs function
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline val Float.absoluteValue: Float get() = nativeMath.abs(this.toDouble()).toFloat()

/**
 * Returns the sign of this value:
 * -1.0 if the value is negative,
 * zero if the value is zero,
 * 1.0 if the value is positive
 * Special case:
 * `sign(NaN)` is `NaN`
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline val Float.sign: Float get() = nativeMath.sign(this.toDouble()).toFloat()

/**
 * Returns this value with the sign bit same as of the [sign] value.
 * If [sign] is `NaN` the sign of the result is undefined.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun Float.withSign(sign: Float): Float = this.toDouble().withSign(sign.toDouble()).toFloat()

/**
 * Returns this value with the sign bit same as of the [sign] value.
 * If [sign] is `NaN` the sign of the result is undefined.
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun Float.withSign(sign: Int): Float = this.toDouble().withSign(sign.toDouble()).toFloat()

/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 * Special cases:
 * - `round(x)` is `x` where `x` is `Int.MAX_VALUE`
 * @throws IllegalArgumentException when this value is `Int.MIN_VALUE` when `x < Int.MIN_VALUE`
 * @since Kotlin("1.2")\n @inlineonly
 * @public actual inline fun Float.roundToInt(): Int =

Open Source Used In DNAC Platform Arica 8125
toDouble().roundToLong() Rounds this [Float] value to the nearest integer and converts the result to
[Long].

Ties are rounded towards positive infinity.

Special cases:

- `x.roundToLong()` == `Long.MAX_VALUE` when `x > Long.MAX_VALUE`
- `-x.roundToLong()` == `Long.MIN_VALUE` when `x < Long.MIN_VALUE`

@throws IllegalArgumentException when this value is `NaN`

*roundToLong*() @InlineOnly

public actual inline fun Float.roundToLong(): Long =
toDouble().roundToLong() // endregion

---

**roundToLong()**

Returns the absolute value of the given value [n].

Special cases:

- `abs(Int.MIN_VALUE)` is `Int.MIN_VALUE` due to an overflow

Valued `Int.absoluteValue` extension property for [Int]

// TODO: remove manual 'or' when KT-19290 is fixed

Valued `Int.sign` extension property for [Int]

---

// endregion

// region ================ Integer Math

/**
 * Returns the absolute value of the given value [n].
 *
 * Special cases:
 *   - `abs(Long.MIN_VALUE)` is `Long.MIN_VALUE` due to an overflow
 *
 * @see absoluteValue extension property for [Long]
 */

@SinceKotlin("1.2")
public actual fun abs(n: Long): Long = if (n < 0) -n else n

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public actual inline fun min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public actual inline fun max(a: Long, b: Long): Long = if (a >= b) a else b

/**
 * Returns the absolute value of this value.
 *
 * Special cases:
 *   - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow
 *
 * @see abs function
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline val Long.absoluteValue: Long get() = abs(this)

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 *   - `0` if the value is zero,
 *   - `1` if the value is positive
 */
@SinceKotlin("1.2")
public actual val Long.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}

---

// endregion

---

/**
 * Returns `true` if the specified number is a Not-a-Number (NaN) value, `false` otherwise.
 */

public actual fun Double.isNaN(): Boolean = this != this

/**
 * Returns `true` if the specified number is a Not-a-Number (NaN) value, `false` otherwise.
 */

public actual fun Float.isNaN(): Boolean = this != this

/**
 * Returns `true` if this value is infinitely large in magnitude.
 */

public actual fun Double.isInfinite(): Boolean = this == Double.POSITIVE_INFINITY || this == Double.NEGATIVE_INFINITY

/**
 * Returns `true` if this value is infinitely large in magnitude.
 */

public actual fun Float.isInfinite(): Boolean = !isNaN() && this != Double.NEGATIVE_INFINITY

---

/**
 * Returns the number of set bits in the binary representation of this [Int] number.
 */

@WasExperimental(ExperimentalStdlibApi::class)

@SinceKotlin("1.4")
public actual fun Int.countOneBits(): Int {
    // Hacker's Delight 5-1 algorithm
    var v = this
    v = (v and 0x55555555) + (v.ushr(1) and 0x55555555)
    v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)
    v = (v and 0x0F0F0F0F) + (v.ushr(4) and 0x0F0F0F0F)
    v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)
0x0000FFFF) + (v.ushr(16))\nreturn \n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("DEPRECATION_ERROR")\n@kotlin.internal.InlineOnly\n\npublic actual inline fun Int.countLeadingZeroBits(): Int =
  kotlin.js.Math.clz32(this)\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Int.countTrailingZeroBits(): Int =
  // Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with countLeadingZeroBits\n  \n  INT.SIZE_BITS - (this or -this).inv().countLeadingZeroBits(\n    this\n  )\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number,\n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Int.takeHighestOneBit(): Int =
  if (this == 0) 0 else 1 shl (INT.SIZE_BITS - 1 - countLeadingZeroBits())\n\n/**\n * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n\npublic actual fun Int.rotateLeft(bitCount: Int): Int =
  shl(bitCount) or ushr(INT.SIZE_BITS - bitCount)\n\n/**\n * Counts the number of set bits in the binary representation of this [Long] number.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Long.countOneBits(): Int =
  high.countOneBits() + low.countOneBits()\n\n/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Long.countLeadingZeroBits(): Int =
  when (val high = this.high) {
    0 -> high.countLeadingZeroBits()\n    else -> high.countLeadingZeroBits()\n  } or \n  when (val low = this.low) {
    0 -> low.countLeadingZeroBits()\n    else -> low.countLeadingZeroBits()\n  }\n\n/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Long.countTrailingZeroBits(): Int =
  when (val high = this.high) {
    0 -> high.countTrailingZeroBits()\n    else -> high.countTrailingZeroBits()\n  } or \n  when (val low = this.low) {
    0 -> low.countTrailingZeroBits()\n    else -> low.countTrailingZeroBits()\n  }\n\n/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,\n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n\npublic actual fun Long.takeHighestOneBit(): Long =
  when (val high = this.high) {
    0 -> Long(0, high.takeHighestOneBit())\n    else -> Long(0, high.takeHighestOneBit())\n  } or \n  when (val low = this.low) {
    0 -> Long(0, low.takeHighestOneBit())\n    else -> Long(0, low.takeHighestOneBit())\n  }

Open Source Used In DNAC Platform Arica  8127
else -> Long(long.takeLowestOneBit(), 0) }\n
  * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.\n  * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n  * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.\n  * number.rotateLeft(-n) == number.rotateRight(n)\n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateLeft(n) == number.rotateLeft(n % 64) \n
  * @SinceKotlin("1.3")\n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * @SinceKotlin("1.3")\n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64) \n
  * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n  * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n  * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.\n  * number.rotateRight(-n) == number.rotateLeft(n) \n
  * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n  * number.rotateRight(n) == number.rotateRight(n % 64)
**ExperimentalAssociatedObjects**

- Makes the annotated annotation class an associated object key.
- An associated object key annotation should have single [KClass] parameter. When applied to a class with reference to an object declaration as an argument, it binds the object to the class, making this binding discoverable at runtime using [findAssociatedObject].

```kotlin
@ExperimentalAssociatedObjects
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.CLASS)
public annotation class AssociatedObjectKey
```

```kotlin
/**
 * If [T] is an @AssociatedObjectKey-annotated annotation class and [this] is annotated with @[T] (S::class), returns object `S`.
 * Otherwise returns `null`.
 */
@ExperimentalAssociatedObjects
public inline fun <reified T : Annotation> KClass<*>.findAssociatedObject(): Any? = this.findAssociatedObject(T::class)
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.js

import getKClass
import kotlin.reflect.KClass
import kotlin.reflect.js.internal.KClassImpl

/**
 * Represents the constructor of a class. Instances of `JsClass` can be passed to JavaScript APIs that expect a constructor reference.
 * *
 * @install
 * @external
 * @interface
 * @class

```kotlin
external interface JsClass<T : Any> {
    /**
     * Returns the unqualified name of the class represented by this instance.
     */
    val name: String
}
```

```kotlin
val <T : Any> KClass<T>.js: JsClass<T>
    get() = (this as KClassImpl<T>).jClass
```

```kotlin
/**
 * Obtains a constructor reference for the given `KClass`.
 */
val <T : Any> JsClass<T>.kotlin: KClass<T>
    get() = getKClass(this)
```

```kotlin
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal abstract class KClassImpl<T : Any>(
    internal open val jClass: JsClass<T>
) : KClass<T> {

    override val qualifiedName: String?
        get() = TODO()//un override fun equals(other: Any?): Boolean { return other is KClassImpl<*> && jClass == other.jClass } // TODO: use FQN override fun hashCode(): Int = simpleName?.hashCode() ?: 0n override fun toString(): String { // TODO: use FQN return \"class \$simpleName\" } //un override fun isInstance(value: Any?): Boolean { return jsIsType(value, jClass) } override fun equals(other: Any?): Boolean = other is SimpleKClassImpl && givenSimpleName == other.givenSimpleName override val givenSimpleName: String = jClass.asDynamic().\$metadata$.simpleName.unsafeCast<String?>() //un override fun isInstance(value: Any?): Boolean { return jsIsType(value, jClass) } //un override fun equals(other: Any?): Boolean = other/private val givenSimpleName: String
```

```kotlin
internal class SimpleKClassImpl<T : Any>(jClass: JsClass<T>) : KClassImpl<T>(jClass) {
    override fun equals(other: Any?): Boolean {
        if (other !is SimpleKClassImpl<*>) return false
        return super.equals(other) && givenSimpleName == other.givenSimpleName
    }

    override val simpleName: String? get() = givenSimpleName

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override val isInstanceFunction: (Any?) -> Boolean
    get() = givenSimpleName

```

```kotlin
internal class PrimitiveKClassImpl<T : Any>(
    jClass: JsClass<T>,
    private val givenSimpleName: String,
    private val isInstanceFunction: (Any?) -> Boolean
) : KClassImpl<T>(jClass) {
    override fun equals(other: Any?): Boolean {
        if (other !is PrimitiveKClassImpl<*>) return false
        return givenSimpleName == other.givenSimpleName && isInstanceFunction(value)
    }

    override val givenSimpleName: String
    get() = givenSimpleName

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override val isInstanceFunction: (Any?) -> Boolean
    get() = givenSimpleName

```

```kotlin
internal object NothingKClassImpl : KClassImpl<Nothing>(js("Object")) {
    override val simpleName: String = "Nothing"

    override fun isInstance(value: Any?): Boolean = false

    override val jClass: JsClass<Nothing>
        get() = throw UnsupportedOperationException("There's no native JS class for Nothing type\")

    override fun equals(other: Any?): Boolean = other === this

    override fun hashCode(): Int = 0
}
```

```kotlin
internal class ErrorKClass : KClass<Nothing> {
    override val simpleName: String? get() = error("Unknown simpleName for ErrorKClass")
    override val qualifiedName: String? get() = error("Unknown qualifiedName for ErrorKClass")

    override fun isInstance(value: Any?): Boolean = error("Can't check isInstance on ErrorKClass")

    override fun equals(other: Any?): Boolean = other === this

    override fun hashCode(): Int = 0
}
```

```kotlin
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect

internal actual inline val KClass<*>.qualifiedOrSimpleName: String?
    get() = simpleName
```
Open Source Used In DNAC Platform Arica 8130

```kotlin
package kotlin.reflect.js.internal

import kotlin.reflect.*

internal data class KTypeParameterImpl(
    override val name: String,
    override val upperBounds: List<KType>,
    override val variance: KVariance,
    override val isReified: Boolean
) : KTypeParameter {
    override fun toString(): String = name
}

internal object DynamicKType : KType {
    override val classifier: KClassifier? = null
    override val arguments: List<KTypeProjection> = emptyList()
    override val isMarkedNullable: Boolean = false
    override fun toString(): String = "dynamic"
}

internal fun KVariance.prefixString() =
    when (this) {
        KVariance.INVARIANT -> ""
        KVariance.IN -> "in ",
        KVariance.OUT -> "out ",
        else -> "KVariance.OUT"
    }

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal class KTypeImpl(
    override val classifier: KClassifier,
    override val arguments: List<KTypeProjection>,
    override val isMarkedNullable: Boolean
) : KType {
    override fun equals(other: Any?): Boolean =
        other is KTypeImpl &&
                classifier == other.classifier && arguments == other.arguments &&
                isMarkedNullable == other.isMarkedNullable

    override fun hashCode(): Int =
        (classifier.hashCode() * 31 + arguments.hashCode()) * 31 + isMarkedNullable.hashCode()

    override fun toString(): String {
        val kClass = (classifier as? KClass<*>)
        val classifierName = when {
            kClass == null -> classifier.toString()
            kClass.simpleName != null -> kClass.simpleName
            else -> "(non-denotable type)"
        }

        val args =
            if (arguments.isEmpty()) ""
            else arguments.joinToString(", ", ", "<", ", ">") { it.asString() }
        val nullable = if (isMarkedNullable) "?" else ""

        return classifierName + args + nullable
    }
}
```

---

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Any").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
}
```

---

```kotlin
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlinx.js as Js

@JsName("PrimitiveClasses")
internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Any").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
}
```

---

```kotlin
import kotlinx.js as Js

@JsName("PrimitiveClasses")
internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Any").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
}
```

---

```kotlin
import kotlinx.js as Js

@JsName("PrimitiveClasses")
internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Any").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
}
```

---

```kotlin
import kotlinx.js as Js

@JsName("PrimitiveClasses")
internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Any").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })
}
```
booleanClass = PrimitiveKClassImpl(js("Boolean"), unsafeCast<JsClass<Boolean>>(), "Boolean", { it is Boolean })
byteClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Byte>>(), "Byte", { it is Byte })
shortClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Short>>(), "Short", { it is Short })
intClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Int>>(), "Int", { it is Int })
floatClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Float>>(), "Float", { it is Float })
doubleClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Double>>(), "Double", { it is Double })
arrayClass = PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<Array<*>>>(), "Array", { it is Array<*> })
stringClass = PrimitiveKClassImpl(js("String"), unsafeCast<JsClass<String>>(), "String", { it is String })
throwableClass = PrimitiveKClassImpl(js("Error"), unsafeCast<JsClass<Throwable>>(), "Throwable", { it is Throwable })
booleanArrayClass = PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<BooleanArray>>(), "BooleanArray", { it is BooleanArray })
charArrayClass = PrimitiveKClassImpl(js("Uint16Array"), unsafeCast<JsClass<CharArray>>(), "CharArray", { it is CharArray })
byteArrayClass = PrimitiveKClassImpl(js("Int8Array"), unsafeCast<JsClass<ByteArray>>(), "ByteArray", { it is ByteArray })
shortArrayClass = PrimitiveKClassImpl(js("Int16Array"), unsafeCast<JsClass<ShortArray>>(), "ShortArray", { it is ShortArray })
ingteArrayClass = PrimitiveKClassImpl(js("Int32Array"), unsafeCast<JsClass<IntArray>>(), "IntArray", { it is IntArray })
longArrayClass = PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<LongArray>>(), "LongArray", { it is LongArray })
floatArrayClass = PrimitiveKClassImpl(js("Float32Array"), unsafeCast<JsClass<FloatArray>>(), "FloatArray", { it is FloatArray })
doubleArrayClass = PrimitiveKClassImpl(js("Float64Array"), unsafeCast<JsClass<DoubleArray>>(), "DoubleArray", { it is DoubleArray })
functionClass = fun functionClass(arity: Int): KClassImpl<Any> {
    @JsName("getKClass")
    internal fun <T : Any>
    getKClass1(jClass: JsClass<T>): KClass<T> = functionClassFromExpression("")

function e -> PrimitiveClasses.functionClass(e.asDynamic().length)
else -> {
    when (e is BooleanArray) -> PrimitiveClasses.booleanArrayClass
    else -> {
        when {
            e is CharArray -> PrimitiveClasses.charArrayClass
            e is ByteArray -> PrimitiveClasses.byteArrayClass
            e is ShortArray -> PrimitiveClasses.shortArrayClass
            e is IntArray -> PrimitiveClasses.intValue
            e is LongArray -> PrimitiveClasses.longArrayClass
            e is FloatArray -> PrimitiveClasses.floatValue
            e is DoubleArray -> PrimitiveClasses.doubleArrayClass
            e is KClass<*> -> KClass::class
            e is Array<*> -> PrimitiveClasses.arrayClass
            else -> {
                val constructor = js("Object").getPrototypeOf(e).constructor
                when {
                    constructor === js("Object") -> PrimitiveClasses.anyClass
                    constructor === js("Error") -> PrimitiveClasses.throwableClass
                    else -> {
                        val jsClass: JsClass<T> = constructor
                        getKClass1(jsClass)
                    }
                }
            }
        }
    }
} unsafeCast<KClass<T>>()

@JsName("getKClass1")
internal fun <T : Any> getKClass1(jClass: JsClass<T>): KClass<T> {
    if (jClass === js("String")) return PrimitiveClasses.stringClass

    val metadata = jClass.asDynamic().`$metadata$`
    return if (metadata != null) {
        if (metadata.`$kClass$` == null) {
            val kClass = SimpleKClassImpl(jClass)
            metadata.`$kClass$` = kClass
            kClass
        } else {
            metadata.`$kClass$`
        }
    } else {
        SimpleKClassImpl(jClass)
    }
}
Open Source Used In DNAC Platform Arica 8134

StringBuilder {
        AbstractList.checkPositionIndex(index, length)

        string = string.substring(0, index) + @WasExperimental(ExperimentalStdlibApi::class)
    actual fun insert(index: Int, value: CharSequence?):
            "null" if [] is `null`, then the four characters `"null"` are inserted.
            *
            * @param value the character sequence from which characters are inserted. If
        string.substring(index)
        return this
    }

    /**
     * Inserts characters in the specified character array
     * into this string builder at the specified [index] and returns this instance.
     *
     * The inserted characters go in the same order as in the [value] character array,
     * starting at [index].
     *
     * @param index the position
     * @param value the character array
     * @param startIndex the size of the backing storage is not extended to comply the given [minimumCapacity],
     * thus calling this method has no effect on the further performance of operations.
        */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun ensureCapacity(minimumCapacity: Int) {
        return this.string.asDynamic().indexOf(string)
    }

    /**
     * Returns the index within this string builder of the last occurrence of the specified [string].
        * The last occurrence of empty string `""` is considered to be at the index equal to `this.length`.
        * Returns `-1` if the specified [string] does not occur in this string builder.
        */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun lastIndexOf(string: String, start: Int): Int =
        Int = this.string.asDynamic().lastIndexOf(string)

    /**
     * Returns the index within this string builder of the last occurrence of the specified [string],
     * starting from the specified [startIndex] toward the beginning.
        * Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].
        */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun lastIndexOf(string: String, startIndex: Int): Int =
        this.string.asDynamic().lastIndexOf(string, startIndex)

    /**
     * Returns the index within this string builder of the first occurrence of the specified [string].
        * Returns `-1` if the specified [string] does not occur in this string builder.
        */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun indexOf(string: String, startIndex: Int): Int =
        this.string.asDynamic().indexOf(string, startIndex)

    /**
     * Returns the index within this string builder of the first occurrence of the specified [string].
            * starting at the specified [startIndexOf].
            * The overall effect is exactly as if the [value] were converted to a string by the `\value.to(String)` method.
            * and then that string was inserted into this string builder at the specified [index].
            * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
            */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun ensureCapacity(minimumCapacity: Int) {
        return this.string.asDynamic().indexOf(string)
    }

    /**
     * Returns the index within this string builder of the first occurrence of the specified [string].
            * The overall effect is exactly as if the [value] were converted to a string by the `\value.to(String)` method.
            * and then that string was inserted into this string builder at the specified [index].
            * @param index the position
            * @param value the character sequence from which characters are inserted. If
            * @param startIndex the size of the backing storage is not extended to comply the given [minimumCapacity],
            * thus calling this method has no effect on the further performance of operations.
            */
        Int = this.string.asDynamic().indexOf(string)

    /**
     * Returns the index within this string builder of the last occurrence of the specified [string].
            * The last occurrence of empty string `""` is considered to be at the index equal to `this.length`.
            * Returns `-1` if the specified [string] does not occur in this string builder.
            */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun lastIndexOf(string: String) =
        this.string.asDynamic().lastIndexOf(string)

    /**
     * Returns the index within this string builder of the last occurrence of the specified [string],
     * starting from the specified [startIndexOf].
     * The last occurrence of empty string `""` is considered to be at the index equal to `this.length`.
     * Returns `-1` if the specified [string] does not occur in this string builder.
     */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun ensureCapacity(minimumCapacity: Int) {
        return this.string.asDynamic().indexOf(string)
    }

    /**
     * Returns the index within this string builder of the first occurrence of the specified [string].
            * Returns `-1` if the specified [string] does not occur in this string builder.
            */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    actual fun indexOf(string: String) =
        this.string.asDynamic().indexOf(string)
value.toString() + string.substring(index)\n    return this\n} /** * Inserts the string representation of
the specified object [value] into this string builder at the specified [index] and returns this instance.\n */ * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method, \n * and then that string was inserted into this string builder at the specified [index].\n */ @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.\n */ @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun insert(index: Int, value: Any?): StringBuilder {\n        StringBuilder {\n            AbstractList.checkPositionIndex(index, length)\n            val toInsert = value ?: "\null\"\n            this.string = this.string.substring(0, index) + toInsert + this.string.substring(index)\n        }\n    }\n
    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun insert(index: Int, value: String?): StringBuilder {\n        AbstractList.checkPositionIndex(index, length)\n        val toInsert = value ?: value.toString() + string.substring(index)\n        return this\n    }\n
    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun set(index: Int, value: Char) {\n        AbstractList.checkElementIndex(index, length)\n        string = string.substring(0, index) + value.toString() + string.substring(index)\n    }\n
    /**\n     * Replaces characters in the\n     * @sample samples.text.Strings.clearStringBuilder\n     */\n    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun clear(): String = string\n
    /**\n     * Sets the character at the specified [index]\n     * @sample samples.text.Strings.clearStringBuilder\n     */\n    @SinceKotlin("1.4")\n    actual fun setLength(newLength: Int) {\n        if (newLength < 0) {\n            throw IllegalArgumentException("Negative new length: \n\n        }\n
        if (newLength <= length) {\n            string = string.substring(0, newLength)\n        }\n
        else {\n            for (i in length until newLength) {\n                string += "\u000000000"\n            }\n        }\n    }\n
    /**\n     * Inserts the string \n\n    }\n
    /**\n     * Sets the length of this string builder to the specified [newLength].\n    */\n    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun substring(startIndex: Int, endIndex: Int): String {\n        AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n        return string.substring(startIndex, endIndex)\n    }\n
    /**\n     * Returns a new [StringBuilder] that contains characters in this string builder at \n     * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of this string builder.\n    */\n    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun trimToSize() {\n    }\n
    override fun toString(): \n\n    public fun clear(): String = string\n    public fun insert(index: Int, value: Any?): StringBuilder = \n\n    */
    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun insert(index: Int, value: Any?): StringBuilder = \n\n    */
    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun set(index: Int, value: Char) {\n        AbstractList.checkElementIndex(index, length)\n        string = string.substring(0, index) + value + string.substring(index + 1)\n    }\n
    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun substring(startIndex: Int, endIndex: Int): String {\n        AbstractList.checkBoundsIndexes(startIndex, endIndex, length)\n        return string.substring(startIndex, endIndex)\n    }\n
    /**\n     * Attempts to reduce storage used for this string builder.\n    */\n    @SinceKotlin("1.4")\n    @WasExperimental(ExperimentalStdlibApi::class)\n    actual fun trimToSize() {\n        }\n
        if (newLength <= length) {\n            string = string.substring(0, newLength)\n        }\n
        else {\n            for (i in length until newLength) {\n                string += "\u000000000"\n            }\n        }\n    }\n
    /**\n     * Inserts the string representation of
specified range of this string builder with characters in the specified string [value] and returns this instance.

* @param startIndex the beginning (inclusive) of the range to replace.
* @param endIndex the end (exclusive) of the range to replace.
* @param value the string to replace with.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.

@WasExperimental(ExperimentalStdlibApi::class)
public fun setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)
    return this
}

private fun checkReplaceRange(startIndex: Int, endIndex: Int, length: Int) {
    if (startIndex < 0 || startIndex > length) {
        throw IndexOutOfBoundsException("startIndex: $startIndex, length: $length")
    }
    if (startIndex > endIndex) {
        throw IllegalArgumentException("startIndex($startIndex) > endIndex($endIndex)
    }
}

/**
 * Removes the character at the specified [index] from this string builder and returns this instance.
 * If the 'Char' at the specified [index] is part of a supplementary code point, this method does not remove
 * the entire supplementary character.
 * @param index the index of 'Char' to remove.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [index] is out of range of this string builder.
 * @throws IndexOutOfBoundsException when the range doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index
 * is out of the [destination] array indices range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @param destination the array indices range.
 * @param destinationOffset the position in the array to copy to, 0 by default.
of the subsequence to append.

    * @param endIndex the end (exclusive) of the subsequence to append.

    * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

    * @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    public fun appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {
        val stringCsq = value.toString()

        AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)

        string += stringCsq.substring(startIndex, endIndex)
        return this
    }

    /**
     * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.

     * The inserted characters go in same order as in the [value] array, starting at [startIndex] and ending at [endIndex].

     * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
     * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.

     * @SinceKotlin("1.4")
     @WasExperimental(ExperimentalStdlibApi::class)
     public fun insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {

        AbstractList.checkPositionIndex(index, this.length)

        @OptIn(ExperimentalStdlibApi::class)
        string = string.substring(0, index) + value.concatToString(startIndex, endIndex) + string.substring(index)
        return this
    }

    /**
     * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.

     * The inserted characters go in same order as in the [value] character sequence, starting at [startIndex] and ending at [endIndex].

     * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
     * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.

     * @SinceKotlin("1.4")
     @WasExperimental(ExperimentalStdlibApi::class)
     public fun insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {

        AbstractList.checkPositionIndex(index, length)

        val stringCsq = value.toString()

        AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)

        string = string.substring(0, index) + stringCsq.substring(startIndex, endIndex) + string.substring(index)
        return this
    }

    /**
     * Clears the content of this string builder making it empty and returns this instance.

     * @sample samples.text.Strings.clearStringBuilder

     */
    @SinceKotlin("1.3")
    @Suppress("EXTENSION_SHADED_BY_MEMBER", "NOTHING_TO_INLINE")
    public actual inline fun StringBuilder.clear(): StringBuilder = this.clear()

    /**
     * Sets the character at the specified [index] to the specified [value].

     * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.

     */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    @Suppress("EXTENSION_SHADED_BY_MEMBER", "NOTHING_TO_INLINE")
    public actual inline operator fun StringBuilder.set(index: Int, value: Char) = this.set(index, value)

    /**
     * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.

     * @param startIndex the beginning (inclusive) of the range to replace.
     * @param endIndex the end (exclusive) of the range to replace.

     * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.

     */
    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    @Suppress("EXTENSION_SHADED_BY_MEMBER", "NOTHING_TO_INLINE")
    public actual inline fun StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder =
        this.setRange(startIndex,
endIndex, value)\n\n* * * Removes the character at the specified [index] from this string builder and returns this instance.\n* * * If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.\n* * * @param index the index of `Char` to remove.\n* * * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n
* * *\n* * *\n* * *\n
```kotlin
StringBuilder = this.deleteAt(index)\n
* * *\n* * *\n* * *\n```

@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.deleteAt(index: Int): StringBuilder = this.deleteAt(index)

```kotlin
StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder = this.deleteRange(startIndex, endIndex)
```

@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder = this.deleteRange(startIndex, endIndex)

```kotlin
StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =\n    this.toCharArray(destination, destinationOffset, startIndex, endIndex)
```

@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =
    this.toCharArray(destination, destinationOffset, startIndex, endIndex)

```kotlin
StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =\n    this.appendRange(value, startIndex, endIndex)
```

@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)

```kotlin
StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =\n    this.appendRange(value, startIndex, endIndex)
```

@SinceKotlin("1.4")\n@WasExperimental(ExperimentalStdlibApi::class)\n@Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)
to insert at.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =
this.insertRange(index, value, startIndex, endIndex)
```

```kotlin
/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =
this.insertRange(index, value, startIndex, endIndex)
```

```kotlin
/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder =
this.insertRange(index, value, startIndex, endIndex)
```

```kotlin
/**
 * Returns `true` if this string is not `null` and its content is equal to the word "true", ignoring case, and `false` otherwise.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 * @param radix the radix to use.
 */
public actual fun String.toBoolean(): Boolean =
this.toBoolean()
```

```kotlin
/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @param radix the radix to use.
 */
public actual fun String.toByte(radix: Int): Byte =
this.toByte(radix)
```

```kotlin
/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @param radix the radix to use.
 */
public actual fun String.toByte(): Byte =
this.toByte()
```
/**
 * Parses the string as a [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toShort(radix: Int): Short = toShortOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an [Int] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toInt(): Int = toIntOrNull() ?: numberFormatError(this)

/**
 * Parses the string as an [Int] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toInt(radix: Int): Int = toIntOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as a [Long] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toLong(): Long = toLongOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Long] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toLong(radix: Int): Long = toLongOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as a [Double] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
@kotlin.internal.InlineOnly
public actual fun String.toDouble(): Double = (+(this.asDynamic())).unsafeCast<Double>().also {
    if (it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank())
        numberFormatError(this)
}

/**
 * Parses the string as a [Float] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalFormatException when [radix] is not a valid radix for string to number conversion.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toFloat(): Float = toDouble().unsafeCast<Float>()

/**
 * Parses the string as a [Double] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
public actual fun String.toDoubleOrNull(): Double? =
(+(this.asDynamic())).unsafeCast<Double>().takeIf {
    !(it.isNaN() && !this.isNaN() || it == 0.0 &&
this.isBlank())
}.let { if (it >= radix) -1 else it }

/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun Byte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Short] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun Short.toString(radix: Int): String =
asDynamic().toString(checkRadix(radix))

private fun String.isNaN(): Boolean = when (this.toLowerCase()) {
    "nan", "+nan", "-nan" -> true
    else -> false
}.let { if (it >= radix) -1 else it }

val numberFormatError = IllegalArgumentException("radix $radix was not in valid range 2..36")

internal actual fun digitOf(char: Char, radix: Int): Int = when {
    char >= '0' && char <= '9' -> char - '0'
    char >= 'A' && char <= 'Z' -> char - 'A' + 10
    char >= 'a' && char <= 'z' -> char - 'a' + 10
    else -> -1
}.let { if (it >= radix) -1 else it }

internal actual fun checkRadix(radix: Int): Int {
    if (radix !in 2..36) {
        throw IllegalArgumentException("radix $radix was not in valid range 2..36")
    }
    return radix
}

private fun runI() { throw IllegalFormatException("\$radix was not in valid range 2..36")

}"
*In multiline mode the expressions `^` and `$` match just after or just before, respectively, a line terminator or the end of the input sequence. *

\* @constructor Creates a regular expression from the specified [pattern] string and the specified set of [options].*

* **Create a compiled regular expression.**

\* public actual constructor(pattern: String, option: RegexOption): Regex **Create a regular expression from the specified [pattern] string and the default options.**

\* public actual constructor(pattern: String): Regex **The pattern string of this regular expression.**

\* public actual val pattern: String **The set of options that were used to create this regular expression.**

\* public actual val options: Set<RegexOption> **Indicates whether the regular expression matches the entire [input].**

\* public actual infix fun matches(input: CharSequence): Boolean **Indicates whether the regular expression can find at least one match in the specified [input].**

\* public actual fun containsMatchIn(input: CharSequence): Boolean **Returns the first match of a regular expression in the [input], beginning at the specified [startIndex].**

\* fun find(input: CharSequence, startIndex: Int = 0): MatchResult? **Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the specified [startIndex].**

\* fun matchEntire(input: CharSequence): MatchResult? **Attempts to match the entire [input] against the pattern.**

\* fun replace(input: CharSequence, replacement: String): String **Replaces all occurrences of this regular expression in the specified [input] string with specified [replacement] expression.**
the result of
* the given function [transform] that takes [MatchResult] and returns a string to be used as a
* replacement for that match.
* A public actual inline fun replace(input: CharSequence, transform: 
(MatchResult) -> CharSequence): String { \n var match = find(input) \n if (match == null) return input.toString() \n var lastStart = 0 
 val length = input.length 
 val sb = StringBuilder(length) \n do { \n val foundMatch = match!! 
 sb.append(input, lastStart, foundMatch.range.start) \n sb.append(transform(foundMatch)) \n lastStart = foundMatch.range.endInclusive + 1 \n } while (lastStart < length && & match != null) \n if (lastStart < length) { \n sb.append(input, lastStart, length) \n } \n return sb.toString() \n } /** * Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression.
* @param replacement A replacement expression that can include substitutions. See
* A public actual fun replaceFirst(input: CharSequence, replacement: String): String { \n val nonGlobalOptions = options.map { it.value }.joinToString(separator = ",") \n return input.toString().nativeReplace(RegExp(pattern, nonGlobalOptions), replacement) \n } /** * Splits the [input] CharSequence around matches of this regular expression.
* @param limit Non-negative value specifying the maximum number of substrings
the string can be split to.
* Zero by default means no limit is set.
* A public actual fun split(input: CharSequence, limit: Int = 0): List<String> { \n val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) } \n val result = mutableListOf<String>() \n var lastStart = 0 \n for (match in matches) { \n result.add(input.subSequence(lastStart, match.range.start).toString()) \n lastStart = match.range.endInclusive + 1 \n } \n return result.add(input.subSequence(lastStart, input.length).toString()) \n } /** * Returns the string representation of this regular expression, namely the [pattern] of this regular
expression.
* A public actual fun toString(): String = nativePattern.toString()
* A actual companion object { 
* fun fromLiteral(literal: String): Regex = Regex(escape(literal)) 
* fun escape(literal: String): String = literal.nativeReplace(patternEscape, "\\$\&") 
* fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, "$\$$\$$\$$\$$") 
* private val patternEscape = RegExp("""
-\\^$*+?.()|\[\]\{\}\]""", "g") 
* private val replacementEscape = RegExp("""\$""", "g") 
}
override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it] }.iterator()

override fun get(index: Int): MatchGroup? = match[index]?.let { MatchGroup(it) }

override fun groupValues(): List<String> = match.map { it.toString() }


private var groupValues_: List<String>? = null

override val groupValues: List<String>
    get() {
        if (groupValues_ == null) {
            groupValues_ = object : AbstractList<String>() {
                override fun get(index: Int): String = match[index] ?: ""
            }
        }
        return groupValues_!!
    }

override fun next(): MatchResult? = this@findNext.findNext(input, if (range.isEmpty()) range.start + 1 else range.endInclusive + 1)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

import kotlin.js.RegExp

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeIndexOf(ch: Char, fromIndex: Int): Int = nativeIndexOf(ch.toString(), fromIndex)

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeLastIndexOf(ch: Char, fromIndex: Int): Int = nativeLastIndexOf(ch.toString(), fromIndex)

/**
 * Returns `true` if this string starts with the specified prefix.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.startsWith(prefix: String, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return nativeStartsWith(prefix, 0)
    else
        return regionMatches(0, prefix, 0, prefix.length, ignoreCase)
}

/**
 * Returns `true` if a substring of this string starting at the specified offset [startIndex] starts with the specified prefix.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.startsWith(prefix: String, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return nativeStartsWith(prefix, startIndex)
    else
        return regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase)
}

/**
 * Returns `true` if this string ends with the specified suffix.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.endsWith(suffix: String, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return nativeEndsWith(suffix)
    else
        return regionMatches(length - suffix.length, suffix, 0, suffix.length, ignoreCase)
}

public fun String.matches(regex: String): Boolean {
    val result = this.match(regex)
    return result != null && result.size != 0
}

public actual fun CharSequence.isNotBlank(): Boolean = length == 0 || (if (this is String) this else this.toString()).matches("^[\s\xA0]+$")

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String?.equals(other: String?, ignoreCase: Boolean = false): Boolean =
    if (this == null)
        other == null
    else if (!ignoreCase)
        this == other
    else
        other != null &&
        this.toLowerCase() ==
        other.toLowerCase()

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharSequence.regionMatches(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int,
    ignoreCase: Boolean = false): Boolean =
    regionMatchesImpl(thisOffset, other, otherOffset, length, ignoreCase)

/**
 * Returns a copy of this string having its first letter titlecased using the rules of the default locale,
 * or the original string if it's empty or already starts with a title case letter.
 * The title case of a character is usually the same as its upper case with several exceptions. See the particular list of characters with the special title case form depends on the underlying platform. See the particular list of characters with the special title case form depends on the underlying platform. */

package kotlin.text

@kotlin.internal.InlineOnly
internal actual inline fun String.toTitleCase(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1)
    else this
}

@kotlin.internal.InlineOnly
internal actual inline fun String.downcaseCase(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1)
    else this
}

@kotlin.internal.InlineOnly
internal actual inline fun String.repeat(n: Int): String {
    require(n >= 0) { "Count 'n' must be non-negative, but was $n." }
    return when (n) {
        0 -> ""
        1 -> this.toString()
        else -> {
            var result = ""
            if (!isEmpty()) {
                var s = this.toString()
                var count = n
                while (true) {
                    if ((count and 1) == 1) {
                        result += s
                    }
                    count = count ushr 1
                    if (result.size == result.length) {
                        break
                    }
                }
            }
            result
        }
    }
}


Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\n* if the sequence is malformed and [throwOnMalformed] is false.\n* @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true.\n*/
    if (index >= endIndex) {
        malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0x0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0x381F8000
}

/**
 * Maximum number of bytes needed to encode a single char.\n* * Code points in `0..0x7F` are encoded in a single byte.\n* * Code points in `0x80..0x7FF` are encoded in two bytes.\n* * Code points in `0x800..0xD7FF` or in `0xE000..0xFFFF` are encoded in three bytes.\n* * Surrogate code points in `0xD800..0xDFFF` are not Unicode scalar values, therefore aren't encoded.\n* * Code points in `0x10000..0x10FFFF` are represented by a pair of surrogate 'Char's and are encoded in four bytes.\n*/
private const val MAX_BYTES_PER_CHAR = 3

/**
 * The byte sequence a malformed UTF-16 char sequence is replaced by.\n*/
private val REPLACEMENT_BYTE_SEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(), 0xBF.toByte(), 0xBD.toByte())

/**
 * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].\n*/
internal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {
    require(startIndex >= 0 && endIndex <= string.length && startIndex <= endIndex)

    val bytes = ByteArray((endIndex - startIndex) * MAX_BYTES_PER_CHAR)
    var byteIndex = 0
    var charIndex = startIndex

    while (charIndex < endIndex) {
        val code = string[charIndex++].toInt()
        when {
            code < 0x80 ->
                bytes[byteIndex++] = code.toByte()
            code < 0x800 -> {
                bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
            }
            code < 0xD800 || code >= 0xE000 -> {
                bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()
                bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
            }
            else -> { // Surrogate char value
                val codePoint = codePointFromSurrogate(string, code, charIndex, endIndex, throwOnMalformed)
                if (codePoint <= 0) {
                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[0]
                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[1]
                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[2]
                } else {
                    bytes[byteIndex++] = ((codePoint shr 18) or 0xF0).toByte()
                    bytes[byteIndex++] = ((codePoint shr 12) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = ((codePoint shr 6) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = ((codePoint and 0x3F) or 0x80).toByte()
                }
            }
        }
    }

    return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
}

/**
 * The character a malformed UTF-8 byte sequence is replaced by.\n*/
private const val REPLACEMENT_CHAR = '\uFFFD'

/**
 * Decodes the UTF-8 [bytes] array and returns the resulting [String].\n*/
internal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {
    require(startIndex >= 0 && endIndex <= bytes.size && startIndex <= endIndex)

    val string = StringBuilder(endIndex - startIndex)
    var byteIndex = startIndex
    while (byteIndex < endIndex) {
        val code = bytes[byteIndex++].toInt()
        when {
            code < 0x80 ->
                string.append((code and 0x7F).toChar())
            code < 0x800 -> {
                string.append(((code shr 6) or 0xC0).toChar())
                string.append((code and 0x3F).toChar())
            }
            code < 0xD800 || code >= 0xE000 -> {
                string.append(((code shr 12) or 0xE0).toChar())
                string.append(((code shr 6) and 0x3F).toChar())
                string.append((code and 0x3F).toChar())
            }
            else -> { // Surrogate char value
                val codePoint = codePointFromSurrogate(bytes, byteIndex, endIndex, throwOnMalformed)
                if (codePoint <= 0) {
                    string.append(REPLACEMENT_CHAR)
                } else {
                    string.append(((codePoint shr 18) or 0xF0).toChar())
                    string.append(((codePoint shr 12) and 0x3F) or 0x80).toChar())
                    string.append(((codePoint shr 6) and 0x3F) or 0x80).toChar())
                    string.append((codePoint and 0x3F) or 0x80).toChar())
                }
            }
        }
    }

    return string.toString()
}
(exclusive) of the array to be encoded.

* @param throwOnMalformed whether to throw on malformed byte
  sequence or replace by the [REPLACEMENT_CHAR].
* @throws CharacterCodingException if the array is
  malformed UTF-8 byte sequence and [throwOnMalformed] is true.
  * An internal fun decodeUtf8(bytes: ByteArray, start: Int, end: Int, throwOnMalformed: Boolean): String {
    require(start <= end && end <= bytes.size) && startIndex <= endIndex
    var byteIndex = startIndex
    val stringBuilder = StringBuilder()
    while (byteIndex < endIndex) {
      val byte = bytes[byteIndex++].toInt()
      when {
        byte >= 0 ->
          stringBuilder.append(byte.toChar())
        byte shr 5 == -2 -> {
          val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)
          if (code <= 0) {
            stringBuilder.append(REPLACEMENT_CHAR)
            byteIndex += -code
          } else {
            stringBuilder.append(code.toChar())
            byteIndex += 1
          }
        byte shr 4 == -2 - {
          val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)
          if (code <= 0) {
            stringBuilder.append(REPLACEMENT_CHAR)
            byteIndex += 2
          } else {
            val high = (code - 0x10000) shr 10 or 0xD800
            val low = (code and 0x3FF) or 0xDC00
            stringBuilder.append(high.toChar())
            stringBuilder.append(low.toChar())
            byteIndex += 3
          }
        byte shr 3 == -2 -> {
          val code = codePointFrom4(bytes, byte, byteIndex, endIndex, throwOnMalformed)
          if (code <= 0) {
            stringBuilder.append(REPLACEMENT_CHAR)
            byteIndex += 3
          } else {
            val high = (code - 0x10000) shr 10 or 0xD800
            val low = (code and 0x3FF) or 0xDC00
            stringBuilder.append(high.toChar())
            stringBuilder.append(low.toChar())
            byteIndex += 3
          }
        } else -> {
          malformed(0, byteIndex, throwOnMalformed)
          stringBuilder.append(REPLACEMENT_CHAR)
        }
      }
      byteIndex = byteIndex + 1
      if (byteIndex == endIndex) break
    }
    return stringBuilder.toString()
  }

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin
 * Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be
 * found in the license/LICENSE.txt file. */

package kotlin

/**
 * Returns the detailed description of this
 * throwable with its stack trace.
 * * The detailed description includes:
 * - the short description (see
 * [Throwable.toString]) of this throwable;
 * - the complete stack trace;
 * - detailed descriptions of the exceptions
 * that were [suppressed][suppressedExceptions] in order to deliver this exception;
 * - the detailed description of
 * each throwable in the [Throwable.cause] chain.
 *
 * * @since Kotlin("1.4")
 * @public actual fun Throwable.stackTraceToString(): String = ExceptionTraceBuilder().buildFor(this)
 * @public actual fun Throwable.printStackTrace() {
 *   console.error(this.stackTraceToString())
 * }
 * @public actual fun Throwable.addSuppressed(exception: Throwable) {
 *   if (this !== exception) {
 *     val suppressed = this.asDynamic()._suppressed.unsafeCast<MutableList<Throwable>?>()
 *     if (suppressed == null) {
 *       this.asDynamic()._suppressed = mutableListOf(exception)
 *     } else {
 *       suppressed.add(exception)
 *     }
 *   }
 * }
 * @public actual val Throwable.suppressedExceptions: List<Throwable>
 *    get() {
 *      return this.asDynamic()._suppressed?.unsafeCast<List<Throwable>>() ?: emptyList()
 *    }
 * 
 * class ExceptionTraceBuilder {
 *   private val target = StringBuilder()
 *   private val visited = Array<Throwable>()
 *   private var topStack: String = ""
 *   private var topStackStart: Int = 0
 *   
 *   fun buildFor(exception: Throwable): String = ExceptionTraceBuilder().buildFor(exception)
 *   fun printStackTrace() {
 *     console.error(this.stackTraceToString())
 *   }
 *   fun addSuppressed(exception: Throwable) {
 *     if (this !== exception) {
 *       val suppressed = this.asDynamic()._suppressed?.unsafeCast<MutableList<Throwable>?>()
 *       if (suppressed == null) {
 *         this.asDynamic()._suppressed = mutableListOf(exception)
 *       } else {
 *         suppressed.add(exception)
 *       }
 *     }
 *   }
 *   fun Throwable.suppressedExceptions: List<Throwable>
 *     get() {
 *       return this.asDynamic()._suppressed?.unsafeCast<List<Throwable>>() ?: emptyList()
 *     }
 *   
 *   companion object {
 *     fun buildFor(exception: Throwable): String = ExceptionTraceBuilder().buildFor(exception)
 *     fun printStackTrace() {
 *       console.error(this.stackTraceToString())
 *     }
 *   }
 * 
 *   private fun Throwable.stackTraceToString(): String = this.stackTraceToString()
 *   private fun Throwable.printStackTrace() {
 *     console.error(this.stackTraceToString())
 *   }
 * 
 *   private fun Throwable.addSuppressed(exception: Throwable) {
 *     if (this !== exception) {
 *       val suppressed = this.asDynamic()._suppressed?.unsafeCast<MutableList<Throwable>?>()
 *       if (suppressed == null) {
 *         this.asDynamic()._suppressed = mutableListOf(exception)
 *       } else {
 *         suppressed.add(exception)
 *       }
 *     }
 *   }
 *   private fun Throwable.suppressedExceptions: List<Throwable>
 *     get() {
 *       return this.asDynamic()._suppressed?.unsafeCast<List<Throwable>>() ?: emptyList()
 *     }
 * }
stack.indexOf(shortInfo).let { if (it < 0) 0 else it + shortInfo.length }\n    if (stackStart == 0) \n        topStackStart = stackStart\n    else \n        { if (topStack.isEmpty()) \n            topStack = stack
        } else \n        {\n            target.append(shortInfo).append("\n")\n            if (topStack.isEmpty()) \n                {\n                    topStack = stack\n                    topStackStart = stackStart\n                } else \n                {\n                    stack = dropCommonFrames(stack, stackStart)\n                }\n            \n            if (indent.isNotEmpty()) \n            {\n                // indent stack, but avoid indenting exception message lines\n                val messageLines = if (stackStart == 0) 0 else 1 + shortInfo.count { c -> c == '\n' }\n            } else \n            {\n                target.append(stack).append("\n")\n            }\n        }\n    }\n
        target.append(stack).append("\n")\n
    val suppressed = suppressedExceptions\n    if (suppressed.isNotEmpty()) \n    {\n        val suppressedIndent = indent + "\n        for (s in suppressed) \n            s.dumpFullTrace(suppressedIndent,\n            "Suppressed: ")\n    }\n
    return true
}
PerformanceTimeSource(val performance: Performance) : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = performance.now()
    override fun toString(): String = "TimeSource(self.performance.now())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = kotlin.js.Date.now()
    override fun toString(): String = "TimeSource(Date.now())"
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import kotlin.js.json
import kotlin.math.*

internal actual fun formatToExactDecimals(value: Double, decimals: Int): String {
    val rounded = if (decimals == 0) { value } else {
        val pow = 10.0.pow(decimals)
        kotlin.js.Math.round(abs(value) * pow) / pow * sign(value)
    }
    return rounded.asDynamic().toFixed(decimals).unsafeCast<String>()
}

internal actual fun formatUpToDecimals(value: Double, decimals: Int): String {
    return value.asDynamic().toLocaleString("en-us", json("maximumFractionDigits" to decimals)).unsafeCast<String>()
}

internal actual fun formatScientific(value: Double): String {
    return value.asDynamic().toExponential(2).unsafeCast<String>()
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlinx.dom

import org.w3c.dom.*
import kotlin.contracts.*

/**
 * Creates a new element with the specified 
[init].
 *
 * The element is initialized with the specified 
[init] function.
 */
@SinceKotlin("1.4")
public fun Document.createElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }
    return createElement(name).apply(init)
}

/**
 * Appends a newly created element with the specified 
[name] to this element.
 *
 * The element is initialized with the specified 
[init] function.
 */
@SinceKotlin("1.4")
public fun Element.appendElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) }
    return ownerDocument!!.createElement(name, init).also {
        appendChild(it)
    }
}

/**
 * Returns true if the element
 * has the given CSS class style in its 'class' attribute
 */
@SinceKotlin("1.4")
fun Element.hasClass(cssClass: String): Boolean = className.matches("^(\s+)+\$cssClass($|\s+.*)".toRegex())

/**
 * Adds CSS class
 * to element. Has no effect if all specified classes are already in class attribute of the element.
 *
 * @return true if at least one class has been added
 */
@SinceKotlin("1.4")
fun Element.addClass(vararg cssClasses: String): Boolean {
    if (cssClasses.any { hasClass(it) }) {
        val toBeRemoved = cssClasses.toSet()
        className = className.trim().split("\\s+".toRegex()).filter { it !in toBeRemoved }.joinToString(" ")
        return true
    }

    return false
}

/**
 * Removes all 
[cssClasses] from element. Has no effect if all specified classes are missing in class attribute of the element.
 *
 * @return true if at least one class has been removed
 */
@SinceKotlin("1.4")
fun Element.removeClass(vararg cssClasses: String): Boolean {
    if (cssClasses.any { hasClass(it) }) {
        val presentClasses = className.trim().split("\\s+".toRegex()).filter { !it.isEmpty() }
        className = buildString {
            append(presentClasses)
        }
        return true
    }

    return false
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

/**
 * Converts the string into a regular expression [Regex] with the default options.
 */
@kotlin.internal.InlineOnly
public inline fun String.toRegex(): Regex = Regex(this)

/**
 * Converts the string into a regular expression [Regex] with the specified single [option].
 */
@kotlin.internal.InlineOnly
public inline fun String.toRegex(option: RegexOption): Regex = Regex(this, option)
regular expression [Regex] with the specified set of [options].

```kotlin
open class StringBuilder
```

Surrogate pairs included in this string builder are treated as single characters. *\n * Therefore, the order of the high-

low surrogates is never reversed. *\n * Note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair: *\n * fun reverse():

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:

```kotlin
@SinceKotlin("1.4")
```

It's important to note that the reverse operation may produce new surrogate pairs that

were unpaired low-surrogates and high-surrogates before the operation. *\n * For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair:

```kotlin
StringBuilder
```

Characters are appended in order, starting at the index 0:

```kotlin
@SinceKotlin("1.4")
```

Otherwise, this method takes no action and simply returns:
equal to "this.length". *

* Returns `-1` if the specified [string] does not occur in this string builder.*

*/

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
fun lastIndexOf(string: String): Int/*

* Returns the index within this string builder of the last occurrence of the specified [string].*

* starting from the specified [startIndex] toward the beginning.*

* Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].*/

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
fun lastIndexOf(string: String, startIndex: Int): Int/*

* Inserts the string representation of the specified boolean [value] into this string builder at the specified*

[index] and returns this instance.*

* The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method,*

* and then that string was inserted into this string builder as a string builder and then updating each character by index may slow down your program.*

* Therefore, increasing length of this string builder and then updating each character by index by a function that has non-constant execution time complexity.*

* If the [newLength] is less than zero or greater than the length of the current string builder, the null characters `null` are*

* inserted. If [value] is `null`, then the four characters `"null"` are inserted.*

*/

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
fun setLength(newLength: Int): StringBuilder/*

* Sets the length of this string builder to the specified [newLength].*

* If the [newLength] is less than the current length, it is changed to the specified [newLength].*

* Otherwise, null characters `\u0000` are appended to this string builder until its length is less than the [newLength].*

* Note that in Kotlin/JS [setLength] operator function has non-constant execution time complexity.*

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
fun setLength(newLength: Int): StringBuilder/*

* Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [endIndex]
Open Source Used In DNAC Platform Arica

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
fun substring(startIndex: Int, endIndex: Int): String

***
* Attempts to reduce storage used for this string builder.

* If the backing storage of this string builder is larger than necessary to hold its current contents, then it may be resized to become more space efficient.

* Calling this method may, but is not required to, affect the value of the [capacity] property.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
fun trimToSize(): String

**
* Clears the content of this string builder making it empty and returns this instance.

* @sample samples.text.Strings.clearStringBuilder
* @SinceKotlin("1.3")
**
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
fun expect fun
StringBuilder.clear(): String

***
* Sets the character at the specified [index] to the specified [value].

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect operator fun
StringBuilder.set(index: Int, value: Char)

**
* Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder

***
* Removes the character at the specified [index] from this string builder and returns this instance.

* If the `Char` at the specified [index] is part of a supplementary code point, this method does not remove the entire supplementary character.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.deleteAt(index: Int): StringBuilder

***
* Removes characters in the specified range from this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder

***
* Copies characters from this string builder into the [destination] character array.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length): CharArray

***
* Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.

* Characters are appended in order, starting at specified [startIndex].

* @param value the array from which characters are appended.

* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.

* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")
* @WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder

***
* Appends a
subsequence of the specified character sequence [value] to this string builder and returns this instance.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder

/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * The inserted characters go in same order as in the [value] array, starting at [index].
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder

/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
 * The inserted characters go in the same order as in the [value] character sequence, starting at [index].
 * @param index the position in this string builder to insert at.
 * @param value the character sequence from which a subsequence is inserted.
 * @param startIndex the beginning (inclusive) of the subsequence to insert.
 * @param endIndex the end (exclusive) of the subsequence to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun
StringBuilder.insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder

@Suppress("EXTENSION_SHADOWED_BY_MEMBER")
@Deprecated("Use append(value: Any?) instead", ReplaceWith("append(value = obj)"), DeprecationLevel.WARNING)
@kotlin.internal.InlineOnly
public inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)

/**
 * Builds new string by populating newly created [StringBuilder] using provided [builderAction] and then converting it to [String].
 */
@kotlin.internal.InlineOnly
public inline fun buildString(builderAction: StringBuilder.() -> Unit): String {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return StringBuilder().apply(builderAction).toString()
}

/**
 * Builds new string by populating newly created [StringBuilder] initialized with the given [capacity] using provided [builderAction] and then converting it to [String].
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun buildString(capacity: Int, builderAction: StringBuilder.() -> Unit): String {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return StringBuilder(capacity).apply(builderAction).toString()
}

/** Appends all arguments to the given [StringBuilder] and then converting it to [String].
 */
public fun StringBuilder.append(vararg value: String?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/** Appends all arguments to the given [StringBuilder].
 */
public fun StringBuilder.append(vararg value: Any?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/** Appends a line feed character (`\n`) to this [StringBuilder].
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(): StringBuilder = append("
")

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`).
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: CharSequence?): StringBuilder = append(value).appendLine()

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`).
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: CharSequence?): StringBuilder = append(value).appendLine()
StringBuilder.appendLine(value: String?): StringBuilder = append(value).appendLine()

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`). */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: Any?): StringBuilder = append(value).appendLine()

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`). */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: CharArray): StringBuilder = append(value).appendLine()

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`). */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun StringBuilder.appendLine(value: Char): StringBuilder = append(value).appendLine()

/** Appends [value] to this [StringBuilder], followed by a line feed character (`\n`). */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
<table>
<thead>
<tr>
<th>Constant</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CULL_FACE_MODE</td>
<td>Int</td>
</tr>
<tr>
<td>FRONT_FACE</td>
<td>Int</td>
</tr>
<tr>
<td>DEPTH_RANGE</td>
<td>Int</td>
</tr>
<tr>
<td>DEPTH_WRITEMASK</td>
<td>Int</td>
</tr>
<tr>
<td>DEPTH_CLEAR_VALUE</td>
<td>Int</td>
</tr>
<tr>
<td>DEPTH_FUNC</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_CLEAR_VALUE</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_FUNC</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_FAIL</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_PASS_DEPTH_FAIL</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_PASS_DEPTH_PASS</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_REF</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_VALUE_MASK</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_WRITE_MODE</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_FUNC</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_FAIL</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_PASS_DEPTH_FAIL</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_PASS_DEPTH_PASS</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_REF</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_VALUE_MASK</td>
<td>Int</td>
</tr>
<tr>
<td>STENCIL_BACK_WRITE_MODE</td>
<td>Int</td>
</tr>
<tr>
<td>VIEWPORT</td>
<td>Int</td>
</tr>
<tr>
<td>SCISSOR_BOX</td>
<td>Int</td>
</tr>
<tr>
<td>COLOR_CLEAR_VALUE</td>
<td>Int</td>
</tr>
<tr>
<td>COLOR_WRITEMASK</td>
<td>Int</td>
</tr>
<tr>
<td>UNPACK_ALIGNMENT</td>
<td>Int</td>
</tr>
<tr>
<td>PACK_ALIGNMENT</td>
<td>Int</td>
</tr>
<tr>
<td>MAX_TEXTURE_SIZE</td>
<td>Int</td>
</tr>
<tr>
<td>MAX_VIEWPORT_DIMS</td>
<td>Int</td>
</tr>
<tr>
<td>SUBPIXEL_BITS</td>
<td>Int</td>
</tr>
<tr>
<td>RED_BITS</td>
<td>Int</td>
</tr>
<tr>
<td>GREEN_BITS</td>
<td>Int</td>
</tr>
<tr>
<td>BLUE_BITS</td>
<td>Int</td>
</tr>
<tr>
<td>ALPHA_BITS</td>
<td>Int</td>
</tr>
<tr>
<td>POLYGON_OFFSET_UNITS</td>
<td>Int</td>
</tr>
<tr>
<td>POLYGON_OFFSET_FACTOR</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_BINDING_2D</td>
<td>Int</td>
</tr>
<tr>
<td>SAMPLE_BUFFERS</td>
<td>Int</td>
</tr>
<tr>
<td>SAMPLE_COVERAGE_VALUE</td>
<td>Int</td>
</tr>
<tr>
<td>SAMPLE_COVERAGE_INVERT</td>
<td>Int</td>
</tr>
<tr>
<td>DONT_CARE</td>
<td>Int</td>
</tr>
<tr>
<td>FASTEST</td>
<td>Int</td>
</tr>
<tr>
<td>GENERATE_MIPMAP_HINT</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_MAG_FILTER</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_MIN_FILTER</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_WRAP_S</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_WRAP_T</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_2D</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_POSITIVE_X</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_NEGATIVE_X</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_POSITIVE_Y</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_NEGATIVE_Y</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_POSITIVE_Z</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_NEGATIVE_Z</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE_CUBE_MAP_TEXTURE_SIZE</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE0</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE1</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE2</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE3</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE4</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE5</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE6</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE7</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE8</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE9</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE10</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE11</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE12</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE13</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE14</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE15</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE16</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE17</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE18</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE19</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE20</td>
<td>Int</td>
</tr>
<tr>
<td>TEXTURE21</td>
<td>Int</td>
</tr>
</tbody>
</table>


```kotlin
val TEXTURE22: Int
val TEXTURE23: Int
val TEXTURE24: Int
val TEXTURE25: Int
val TEXTURE26: Int
val TEXTURE27: Int
val TEXTURE28: Int
val TEXTURE29: Int
val TEXTURE30: Int
val TEXTURE31: Int
val ACTIVE_TEXTURE: Int
val REPEAT: Int
val CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int
val FLOAT_VEC3: Int
val FLOAT_VEC4: Int
val INT_VEC2: Int
val INT_VEC3: Int
val INT_VEC4: Int
val BOOL: Int
val BOOL_VEC2: Int
val BOOL_VEC3: Int
val BOOL_VEC4: Int
val SAMPLER_2D: Int
val SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_STRIDE: Int
val VERTEX_ATTRIB_ARRAY_TYPE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val LOW_FLOAT: Int
val MEDIUM_FLOAT: Int
val HIGH_FLOAT: Int
val LOW_INT: Int
val MEDIUM_INT: Int
val HIGH_INT: Int
val FRAMEBUFFER: Int
val RENDERBUFFER: Int
val RGBA4: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT16: Int
val STENCIL_INDEX: Int
val STENCIL_INDEX8: Int
val DEPTH_STENCIL: Int
val RENDERBUFFER_WIDTH: Int
val RENDERBUFFER_HEIGHT: Int
val RENDERBUFFER_INTERNAL_FORMAT: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val CONTEXT_LOST_WEBGL: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
}```
Int
        val ONE_MINUS_CONSTANT_ALPHA: Int
        val BLEND_COLOR: Int
        val
        ARRAY_BUFFER: Int
        val ELEMENT_ARRAY_BUFFER: Int
        val
        ARRAY_BUFFER_BINDING: Int
        val STREAM_DRAW: Int
        val
        STATIC_DRAW: Int
        val DYNAMIC_DRAW: Int
        val BUFFER_SIZE: Int
        val
        BUFFER_USAGE: Int
        val CURRENT_VERTEX_ATTRIB: Int
        val
        FRONT: Int
        val BACK: Int
        val
        FRONT_AND_BACK: Int
        val CULL_FACE: Int
        val
        BLEND_COLOR: Int
        val BLEND: Int
        val
        DITHER: Int
        val STENCIL_TEST: Int
        val DEPTH_TEST: Int
        val
        SCISSOR_TEST: Int
        val
        POLYGON_OFFSET_FILL: Int
        val SAMPLE_ALPHA_TO_COVERAGE: Int
        val
        SAMPLE_COVERAGE: Int
        val
        NO_ERROR: Int
        val INVALID_ENUM: Int
        val
        INVALID_VALUE: Int
        val INVALID_OPERATION: Int
        val
        OUT_OF_MEMORY: Int
        val CW: Int
        val CCW: Int
        val
        LINE_WIDTH: Int
        val
        ALIASED_POINT_SIZE_RANGE: Int
        val
        ALIASED_LINE_WIDTH_RANGE: Int
        val
        CULL_FACE_MODE: Int
        val
        DEPTH_RANGE: Int
        val
        DEPTH_WRITEMASK: Int
        val
        DEPTH_CLEAR_VALUE: Int
        val
        DEPTH_FUNC: Int
        val
        STENCIL_CLEAR_VALUE: Int
        val
        STENCIL_FUNC: Int
        val
        STENCIL_FAIL: Int
        val
        STENCIL_PASS_DEPTH_FAIL: Int
        val
        STENCIL_PASS_DEPTH_PASS: Int
        val
        STENCIL_REF: Int
        val
        STENCIL_VALUE_MASK: Int
        val
        STENCIL_BACK_FUNC: Int
        val
        STENCIL_BACK_FAIL: Int
        val
        STENCIL_BACK_PASS_DEPTH_FAIL: Int
        val
        STENCIL_BACK_PASS_DEPTH_PASS: Int
        val
        STENCIL_BACK_REF: Int
        val
        STENCIL_BACK_VALUE_MASK: Int
        val
        STENCIL_BACK_WRITEMASK: Int
        val
        VIEWPORT: Int
        val
        SCISSOR_BOX: Int
        val
        COLOR_CLEAR_VALUE: Int
        val
        COLOR_WRITEMASK: Int
        val
        UNPACK_ALIGNMENT: Int
        val
        PACK_ALIGNMENT: Int
        val
        MAX_TEXTURE_SIZE: Int
        val
        MAX_VIEWPORT_DIMS: Int
        val
        SUBPIXEL_BITS: Int
        val
        RED_BITS: Int
        val
        GREEN_BITS: Int
        val
        BLUE_BITS: Int
        val
        ALPHA_BITS: Int
        val
        DEPTH_BITS: Int
        val
        STENCIL_BITS: Int
        val
        POLYGON_OFFSET_UNITS: Int
        val
        POLYGON_OFFSET_FACTOR: Int
        val
        TEXTURE_BINDING_2D: Int
        val
        SAMPLE_BUFFERS: Int
        val
        SAMPLES: Int
        val
        SAMPLE_COVERAGE_VALUE: Int
        val
        SAMPLE_COVERAGE_INVERT: Int
        val
        COMPRESSED_TEXTURE_FORMATS: Int
        val
        DONT_CARE: Int
        val
        FASTEST: Int
        val
        NICEST: Int
        val
        GENERATE_MIPMAP_HINT: Int
        val
        BYTE: Int
        val
        UNSIGNED_BYTE: Int
        val
        SHORT: Int
        val
        UNSIGNED_SHORT: Int
        val
        INT: Int
        val
        UNSIGNED_INT: Int
        val
        FLOAT: Int
        val
        DEPTH_COMPONENT: Int
        val
        ALPHA: Int
        val
        RGB: Int
        val
        RGBA: Int
        val
        LUMINANCE: Int
        val
        LUMINANCE_ALPHA: Int
        val
        UNSIGNED_SHORT_4_4_4_4: Int
        val
        UNSIGNED_SHORT_5_5_5_1: Int
        val
        MAX_VERTEX_ATTRIBS: Int
        val
        MAX_VERTEX_UNIFORM_VECTORS: Int
        val
        MAX_VARYING_VECTORS: Int
        val
        MAX_COMBINED_TEXTURE_IMAGE_UNITS: Int
        val
        MAX_TEXTURE_IMAGE_UNITS: Int
        val
        MAX_FRAGMENT_UNIFORM_VECTORS: Int
        val
        SHADERS_TYPE: Int
        val
        DELETE_STATUS: Int
        val
        LINK_STATUS: Int
        val
        VALIDATE_STATUS: Int
        val
        ATTACHED_SHADERS: Int
        val
        ACTIVE_UNIFORMS: Int
        val
        ACTIVE_ATTRIBUTES: Int
        val
        SHADING_LANGUAGE_VERSION: Int
        val
        CURRENT_PROGRAM: Int
        val
        NEVER: Int
        val
        LESS: Int
        val
        EQUAL: Int
        val
        LEQUAL: Int
        val
        GREATER: Int
        val
        NOTEQUAL: Int
        val
        ALWAYS: Int
        val
        KEEP: Int
        val
        REPLACE: Int
        val
        INCR: Int
        val
        DECR: Int
        val
        INVERT: Int
        val
        INCR_WRAP: Int
        val
        DECR_WRAP: Int
        val
        VENDOR: Int
        val
        RENDERER: Int
        val
        VERSION: Int
        val
        NEAREST: Int
        val
        LINEAR: Int
        val
        NEAREST_MIPMAP_NEAREST: Int
        val
        LINEAR_MIPMAP_NEAREST: Int
        val
        TEXTURE_MAG_FILTER: Int
        val
        TEXTURE_MIN_FILTER: Int
        val
        TEXTURE_WRAP_S: Int
        val
        TEXTURE_WRAP_T: Int

Open Source Used In DNAC Platform Arica  8159
val TEXTURE_WRAP_T: Int
val TEXTURE_2D: Int
val TEXTURE: Int
val TEXTURE_CUBE_MAP: Int
val TEXTURE_BINDING_CUBE_MAP: Int
val TEXTURE_CUBE_MAP_POSITIVE_X: Int
val TEXTURE_CUBE_MAP_NEGATIVE_X: Int
val TEXTURE_CUBE_MAP_POSITIVE_Y: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
val TEXTURE_CUBE_MAP_POSITIVE_Z: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int
val MAX_CUBE_MAP_TEXTURE_SIZE: Int
val TEXTURE0: Int
val TEXTURE1: Int
val TEXTURE2: Int
val TEXTURE3: Int
val TEXTURE4: Int
val TEXTURE5: Int
val TEXTURE6: Int
val TEXTURE7: Int
val TEXTURE8: Int
val TEXTURE9: Int
val TEXTURE10: Int
val TEXTURE11: Int
val TEXTURE12: Int
val TEXTURE13: Int
val TEXTURE14: Int
val TEXTURE15: Int
val TEXTURE16: Int
val TEXTURE17: Int
val TEXTURE18: Int
val TEXTURE19: Int
val TEXTURE20: Int
val TEXTURE21: Int
val TEXTURE22: Int
val TEXTURE23: Int
val TEXTURE24: Int
val TEXTURE25: Int
val TEXTURE26: Int
val TEXTURE27: Int
val TEXTURE28: Int
val TEXTURE29: Int
val TEXTURE30: Int
val TEXTURE31: Int
val ACTIVE_TEXTURE: Int
val REPEAT: Int
val CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int
val FLOAT_VEC3: Int
val FLOAT_VEC4: Int
val INT_VEC2: Int
val INT_VEC3: Int
val INT_VEC4: Int
val BOOL: Int
val BOOL_VEC2: Int
val BOOL_VEC3: Int
val BOOL_VEC4: Int
val FLOAT_MAT2: Int
val FLOAT_MAT3: Int
val FLOAT_MAT4: Int
val SAMPLER_2D: Int
val SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_STRIDE: Int
val VERTEX_ATTRIB_ARRAY_TYPE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val COMPARE_FUNC: Int
val EQUAL: Int
val NOTEQUAL: Int
val GREATER: Int
val LESS: Int
val greaterEqual: Int
val lessEqual: Int
val MINUS: Int
val PLUS: Int
val MINUS_REVERSE: Int
val PLUS_REVERSE: Int
val MINUS_CLAMP: Int
val PLUS_CLAMP: Int
val MINUS_CLAMP_TO_EDGE: Int
val PLUS_CLAMP_TO_EDGE: Int
val lessThan: Int
val greaterThan: Int
val lessThanEqual: Int
val greaterThanEqual: Int
val NEVER: Int
val ALWAYS: Int
val LESS: Int
val LESS_EQUAL: Int
val GREATER: Int
val GREATER_EQUAL: Int
val LINEAR: Int
val LINEAR_MIPMAP_LINEAR: Int
val LINEAR_MIPMAP_NEAREST: Int
val NEAREST: Int
val NEAREST_MIPMAP_LINEAR: Int
val NEAREST_MIPMAP_NEAREST: Int
val RGB: Int
val RGBA: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT: Int
val DEPTH_STENCIL: Int
val RGB: Int
val RGBA: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_STENCIL: Int
val TWO_SIDE: Int
val CULL_FACE: Int
val CULL_FACE_MODE: Int
val FILL: Int
val LINE: Int
val LINE_WIDTH: Int
val LINE_WIDTH_CONSERVATIVE: Int
val LINE_WIDTH_NORMALIZED: Int
val LUMINANCE: Int
val LUMINANCE_ALPHA: Int
val LUMINANCE_REQUESTED: Int
val LUMINANCE_EXTERNALLY_REQUESTED: Int
val LUMINANCE8_ALPHA8: Int
val LUMINANCE10_ALPHA10: Int
val LUMINANCE16: Int
val LUMINANCE16_EXT: Int
val LUMINANCE24: Int
val LUMINANCE32F: Int
val LUMINANCE32F_EXT: Int
val LUMINANCE AS SIGNED: Int
val LUMINANCE8: Int
val LUMINANCE8_ALPHA8_EXT: Int
val LUMINANCE8_BIASED: Int
val LUMINANCE8_ALPHA8_BIASED: Int
val LUMINANCE8_ALPHA8_OES: Int
val LUMINANCE16_A16: Int
val LUMINANCE16_ALPHA16: Int
val LUMINANCE24_ALPHA8: Int
val LUMINANCE32F_SGIX: Int
val LUMINANCE8_ALPHA8_BLOCK: Int
val LUMINANCE8_ALPHA8_SPLIT: Int
val LUMINANCE24_8: Int
val LUMINANCE16_16: Int
val LUMINANCE16_16_BLOCK: Int
val LUMINANCE16_16_SPLIT: Int
val LUMINANCE16_16_OES: Int
val LUMINANCE16_16_EXT: Int
val LUMINANCE32F_SGIX_FIXED: Int
val LUMINANCE32F_SGIX_FLOAT: Int
val LUMINANCE32F_SGIX_FLOAT_OES: Int
val LUMINANCE32F_SGIX_FLOAT_EXT: Int
val LUMINANCE2_10_10_10F: Int
val LUMINANCE2_10_10_10F_EXT: Int
val LUMINANCE5_6_5_5F: Int
val LUMINANCE5_6_5_5F_EXT: Int
val LUMINANCE5_11_11_11F: Int
val LUMINANCE5_11_11_11F_EXT: Int
val LUMINANCE8_12: Int
val LUMINANCE8_12_OES: Int
val LUMINANCE16_16_16_16F: Int
val LUMINANCE16_16_16_16F_EXT: Int
val LUMINANCE16_16_16_16F_OES: Int
val LUMINANCE8_16: Int
val LUMINANCE8_16_OES: Int
val LUMINANCE8_16_EXT: Int
val LUMINANCE16_16_16_16F_SGIX: Int
val LUMINANCE16_16_16_16F_SGIX_FIXED: Int
val LUMINANCE8_16_16_16F_SGIX: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_OES: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_EXT: Int
val LUMINANCE16_16_16_16F_SGIX_FIXED: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_SGIX_FIXED: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_SGIX_FLOAT: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_SGIX_FLOAT_OES: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_SGIX_FLOAT_EXT: Int
val LUMINANCE16_16_16_16F_SGIX_FLOAT_SGIX_FLOAT_SGIX_FI...
definedExternally) : Event {
    open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WebGLContextEventInit : EventInit {
    var statusMessage: String?

    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline
    val o = js("({})")
    o["statusMessage"] = statusMessage
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript

[ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin
*/

public external open class ArrayBuffer(length: Int) : BufferDataSource {
    open val byteLength: Int
    fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer

    companion object {
        fun isView(value: Any?): Boolean
    }
}

/**
 * Exposes the JavaScript

[ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin
*/

public external interface ArrayBufferView : BufferDataSource {
    val buffer: ArrayBuffer
    val byteOffset: Int
    val byteLength: Int
}

/**
 * Exposes the JavaScript

[Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin
*/

public external open class Int8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Int8Array)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int8Array, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int8Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator
fun Int8Array.get(index: Int): Byte = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
fun Int8Array.set(index: Int, value: Byte) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript

[Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array) to Kotlin
*/

public external open class Uint8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Uint8Array)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Uint8Array, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint8Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator
fun Uint8Array.get(index: Int): Byte = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
fun Uint8Array.set(index: Int, value: Byte) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript

[Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin
*/

public external open class Int16Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Int16Array)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int16Array, offset: Int = definedExternally)
    fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int16Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator
fun Int16Array.get(index: Int): Short = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
fun Int16Array.set(index: Int, value: Short) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript

[Uint8ClampedArray](https://developer.mozilla.org/en/docs/Web/API/Uint8ClampedArray) to Kotlin
*/

public external open class Uint8ClampedArray : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Uint8ClampedArray)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Uint8ClampedArray, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint8ClampedArray

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator
fun Uint8ClampedArray.get(index: Int): Byte = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
fun Uint8ClampedArray.set(index: Int, value: Byte) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript

[Int16ClampedArray](https://developer.mozilla.org/en/docs/Web/API/Int16ClampedArray) to Kotlin
*/

public external open class Int16ClampedArray : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Int16ClampedArray)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int16ClampedArray, offset: Int = definedExternally)
    fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int16ClampedArray

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator
fun Int16ClampedArray.get(index: Int): Short = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
fun Int16ClampedArray.set(index: Int, value: Short) { asDynamic()[index] = value }
asDynamic() \n
```kotlin
@kotlin.internal.InlineOnly
public inline operator fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }
```

```kotlin
/**
 * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin
 */
public external open class DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) :
    ArrayBufferView {
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun getInt8(byteOffset: Int): Byte
    fun getUint8(byteOffset: Int): Byte
    fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
    fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double
    fun setInt8(byteOffset: Int, value: Byte)
    fun setUint8(byteOffset: Int, value: Byte)
    fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
    fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
    fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally)
    fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally)
}
```

```kotlin
public external interface BufferDataSource
```

```kotlin
/**
 * Exposes the JavaScript [Clipboard](https://developer.mozilla.org/en/docs/Web/API/Clipboard) to Kotlin
 */
public external abstract class Clipboard : EventTarget {
    fun read(): Promise<DataTransfer>
    fun readText(): Promise<String>
    fun write(data: DataTransfer): Promise<Unit>
    fun writeText(data: String): Promise<Unit>
}
```

```kotlin
public external interface ClipboardPermissionDescriptor {
    var allowWithoutGesture: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun ClipboardPermissionDescriptor(allowWithoutGesture: Boolean? = false): ClipboardPermissionDescriptor {
    val o = js("({})")
    o["allowWithoutGesture"] = allowWithoutGesture
    return o
}
```

```kotlin
// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details
```

```kotlin
package org.w3c.dom.clipboard

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface ClipboardEventInit : EventInit {
    var clipboardData: DataTransfer? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["clipboardData"] = clipboardData
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```

```kotlin
/**
 * Exposes the JavaScript [ClipboardEvent](https://developer.mozilla.org/en/docs/Web/API/ClipboardEvent) to Kotlin
 */
public external open class ClipboardEvent(type: String, eventInitDict: ClipboardEventInit = definedExternally) : Event {
    open val clipboardData: DataTransfer?
}
```

```kotlin
/**
 * Exposes the JavaScript [ClipboardPermissionDescriptor](https://developer.mozilla.org/en/docs/Web/API/ClipboardPermissionDescriptor) to Kotlin
 */
public external interface ClipboardPermissionDescriptor {
    var allowWithoutGesture: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details
```
org.w3c.dom.clipboard.*\nimport org.w3c.dom.encleratedmedia.*\nimport org.w3c.dom.events.*\nimport org.w3c.dom.mediacapture.*\nimport org.w3c.dom.mediasource.*\nimport org.w3c.dom.parsing.*\nimport org.w3c.dom.pointerevents.*\nimport org.w3c.dom.svg.*\nimport org.w3c.dom.url.*\nimport org.w3c.fetch.*\nimport org.w3c.files.*\nimport org.w3c.notifications.*\nimport org.w3c.performance.*\nimport org.w3c.workers.*\nimport org.w3c.xhr.*\n
public external abstract class MediaList : ItemArrayLike<String> {
    open var mediaText: String
    fun appendMedium(medium: String)
    fun deleteMedium(medium: String)

    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun MediaList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin
 */
public external abstract class StyleSheet {
    open val type: String
    open val href: String?
    open val ownerNode: UnionElementOrProcessingInstruction?
    open val parentStyleSheet: StyleSheet?
    open val title: String?

    open val media: MediaList
    open var disabled: Boolean
}

/**
 * Exposes the JavaScript
 [CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin
 */
public external abstract class CSSStyleSheet : StyleSheet {
    open val ownerRule: CSSRule?
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)
}

/**
 * Exposes the JavaScript
 [StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin
 */
public external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {
    override fun item(index: Int): StyleSheet?
}

@kotlin.internal.InlineOnly
public inline operator fun StyleSheetList.get(index: Int): StyleSheet? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [LinkStyle](https://developer.mozilla.org/en/docs/Web/API/LinkStyle) to Kotlin
 */
public external interface LinkStyle {
    val sheet: StyleSheet?
        get() = definedExternally
}

/**
 * Exposes the JavaScript
 [CSSRuleList](https://developer.mozilla.org/en/docs/Web/API/CSSRuleList) to Kotlin
 */
public external abstract class CSSRuleList : ItemArrayLike<CSSRule> {
    override fun item(index: Int): CSSRule?
}

@kotlin.internal.InlineOnly
public inline operator fun CSSRuleList.get(index: Int): CSSRule? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [CSSRule](https://developer.mozilla.org/en/docs/Web/API/CSSRule) to Kotlin
 */
public external abstract class CSSRule {
    open val type: Short
    open var cssText: String
    open val parentRule: CSSRule?
    open val parentStyleSheet: CSSStyleSheet?

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 [CSSStyleRule](https://developer.mozilla.org/en/docs/Web/API/CSSStyleRule) to Kotlin
 */
public external abstract class CSSStyleRule : CSSRule {
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 [CSSImportRule](https://developer.mozilla.org/en/docs/Web/API/CSSImportRule) to Kotlin
 */
public external abstract class CSSImportRule : CSSRule {
    open val href: String
    open val medium: MediaList
    open val styleSheet: CSSStyleSheet

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 [CSSGroupingRule](https://developer.mozilla.org/en/docs/Web/API/CSSGroupingRule) to Kotlin
 */
public external abstract class CSSGroupingRule : CSSRule {
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 [CSSMediaRule](https://developer.mozilla.org/en/docs/Web/API/CSSMediaRule) to Kotlin
 */
public external abstract class CSSMediaRule : CSSRule {
    open val media: MediaList
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}
abstract class CSSMediaRule : CSSGroupingRule {
    open val media: MediaList

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSPageRule](https://developer.mozilla.org/en/docs/Web/API/CSSPageRule) to Kotlin
 */
public external abstract class CSSPageRule : CSSGroupingRule {
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSMarginRule](https://developer.mozilla.org/en/docs/Web/API/CSSMarginRule) to Kotlin
 */
public external abstract class CSSMarginRule : CSSRule {
    open val name: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSNamespaceRule](https://developer.mozilla.org/en/docs/Web/API/CSSNamespaceRule) to Kotlin
 */
public external abstract class CSSNamespaceRule : CSSRule {
    open val namespaceURI: String
    open val prefix: String

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript [CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin
 */
public external abstract class CSSStyleDeclaration : ItemArrayLike<String> {
    open var cssText: String
    open var parentRule: CSSRule?
    open var cssFloat: String
    open var alignItems: String
    open var alignSelf: String
    open var animation: String
    open var animationDelay: String
    open var animationDirection: String
    open var animationDuration: String
    open var animationFillMode: String
    open var animationIterationCount: String
    open var animationName: String
    open var animationPlayState: String
    open var animationTimingFunction: String
    open var backfaceVisibility: String
    open var background: String
    open var backgroundColor: String
    open var backgroundImage: String
    open var backgroundClip: String
    open var backgroundPosition: String
    open var backgroundSize: String
    open var border: String
    open var borderBottom: String
    open var borderBottomColor: String
    open var borderBottomLeftRadius: String
    open var borderBottomRightRadius: String
    open var borderBottomStyle: String
    open var borderBottomWidth: String
    open var borderCollapse: String
    open var borderColor: String
    open var borderImage: String
    open var borderImageOutset: String
    open var borderImageRepeat: String
    open var borderImageSlice: String
    open var borderImageSource: String
    open var borderImageWidth: String
    open var borderLeft: String
    open var borderLeftColor: String
    open var borderLeftStyle: String
    open var borderLeftWidth: String
    open var borderRadius: String
    open var borderRight: String
    open var borderRightColor: String
    open var borderRightStyle: String
    open var borderRightWidth: String
    open var borderTop: String
    open var borderTopColor: String
    open var borderTopLeftRadius: String
    open var borderTopRightRadius: String
    open var borderTopStyle: String
    open var borderTopWidth: String
    open var boxDecorationBreak: String
    open var boxSizing: String
    open var boxShadow: String
    open var breakAfter: String
    open var breakBefore: String
    open var breakInside: String
    open var captionSide: String
    open var clear: String
    open var clip: String
    open var color: String
    open var columnCount: String
    open var columnGap: String
    open var columnRuleColor: String
    open var columnRuleStyle: String
    open var columnRuleWidth: String
    open var columnSpan: String
    open var columnWidth: String
    open var content: String
    open var counterIncrement:
String
open var counterReset: String
open var cursor: String
open var direction: String
open var display: String
open var emptyCells: String
open var filter: String
open var flex: String
open var flexBasis: String
open var flexDirection: String
open var flexFlow: String
open var flexGrow: String
open var flexShrink: String
open var flexWrap: String
open var font: String
open var fontFamily:
String
open var fontFeatureSettings: String
open var fontKerning: String
open var fontWeight: String
open var fontSize: String
open var fontSizeAdjust: String
open var fontStyle: String
open var fontStretch: String
open var fontStyle: String
open var fontSynthesis: String
open var fontVariant:
String
open var fontVariantAlternates: String
open var fontVariantCaps: String
open var fontVariantEastAsian: String
open var fontVariantLigatures: String
open var fontVariantNumeric: String
open var fontVariantPosition: String
open var fontWeight: String
open var hangingPunctuation: String
open var height: String
open var hyphens: String
open var imageOrientation: String
open var imageRendering: String
open var imageResolution: String
open var imeMode: String
open var justifyContent: String
open var letterSpacing: String
open var lineHeight: String
open var listStyle: String
open var listStyleImage: String
open var listStylePosition: String
open var listStyleType: String
open var margin: String
open var marginBottom:
String
open var marginLeft:
String
open var marginRight:
String
open var mark: String
open var markAfter: String
open var markBefore: String
open var marks:
String
open var marqueeDirection: String
open var marqueePlayCount: String
open var marqueeSpeed: String
open var marqueeStyle: String
open var mask: String
open var maskType: String
open var maxHeight:
String
open var maxWidth:
String
open var minHeight:
String
open var minWidth:
String
open var navDown:
String
open var navIndex:
String
open var navLeft:
String
open var navRight:
String
open var overflowWrap:
String
open var overflowX:
String
open var overflowY:
String
open var padding:
String
open var paddingBottom:
String
open var paddingLeft:
String
open var paddingRight:
String
open var paddingTop:
String
open var paddingTop:
String
open var pageBreakAfter:
String
open var pageBreakBefore:
String
open var pageBreakInside:
String
open var perspective: String
open var perspectiveOrigin:
String
open var phonemes:
String
open var position:
String
open var quotes:
String
open var rest:
String
open var restAfter:
String
open var restBefore:
String
open var right:
String
open var tabSize:
String
open var tableLayout:
String
open var textAlign:
String
open var textAlignLast:
String
open var textCombineUpright:
String
open var textDecoration:
String
open var textDecorationLine:
String
open var textDecorationStyle:
String
open var textIndent:
String
open var textJustify:
String
open var textOrientation:
String
open var textOverflow:
String
open var textShadow:
String
open var textTransform:
String
open var transitionComment:
String
open var transitionDirection:
String
open var transitionDelay:
String
open var transitionDuration:
String
open var transitionProperty:
String
open var transitionTimingFunction:
String
open var unicodeBidi:
String
open var verticalAlign:
String
open var visibility:
String
open var voiceBalance:
String
open var voiceDuration:
String
open var voicePitch:
String
open var voicePitchRange:
String
open var voiceRate:
String
open var voiceStress:
String
open var voiceVolume:
String
open var whiteSpace:
String
open var wordBreak:
String
open var wordSpacing:
String
open var wordWrap:
String
open var writingMode:
String
open var zIndex:
String
open var _dashed_attribute:
String
open var _camel_cased_attribute:
String
open var _webkit_cased_attribute:
String
open var __webkit_cased_attribute:
String
fun getPropertyIndex(property: String): String
fun getPropertyValue(property: String): String
fun getOwnPropertyDescriptor(property: String): String
fun setProperty(property: String, value: String, priority: String = definedExternally)
fun setPropertyIndex(property: String, value: String, priority: String)
fun setPropertyPriority(property: String, value: String, priority: String)
fun removeProperty(property: String): String
fun override fun item(index: Int):
openSourceUsedInDNACPlatformArica
fun remove(): Promise<Unit>

public external abstract class MediaKeyStatusMap {
    open func size(keyId: String) : Boolean
    fun get(keyId: String): Any?
}

public external open class MediaKeyMessageEvent(type: String, eventInitDict: MediaKeyMessageEventInit) : Event {
    open var messageType: MediaKeyMessageType?
    get() = definedExternally
    set(value) = definedExternally
    var message: ArrayBuffer
    get() = undefined
    set(value) = undefined
}

public external open class MediaKeyStatusMap {
    open val size: Int
    fun has(keyId: String): Boolean
    fun get(keyId: String): Any?
}

public external open class MediaKeyMessageEventEventInit : EventInit {
    open val messageType: MediaKeyMessageType
    open val message: ArrayBuffer

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaKeyMessageEventInit : EventInit {
    var messageType: MediaKeyMessageType?
        get() = definedExternally
        set(value) = definedExternally
    var message: ArrayBuffer?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["messageType"] = messageType
    o["message"] = message
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external open class MediaEncryptedEvent(type: String, eventInitDict: MediaEncryptedEventInit = definedExternally) : Event {
    open val initDataType: String
    open val initData: ArrayBuffer

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaEncryptedEventInit : EventInit {
    var initDataType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var initData: ArrayBuffer? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["initDataType"] = initDataType
    o["initData"] = initData
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaKeysRequirement {
    companion object
}


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaKeySessionType {
    companion object
}

public inline val MediaKeySessionType.Companion.TEMPORARY: MediaKeySessionType get() = "temporary".asDynamic().unsafeCast<MediaKeySessionType>()

public inline val MediaKeySessionType.Companion.PERSISTENT_LICENSE: MediaKeySessionType get() = "persistent-license".asDynamic().unsafeCast<MediaKeySessionType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaKeyStatus {
    companion object
}

public inline val MediaKeyStatus.Companion.USABLE: MediaKeyStatus get() = "usable".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyStatus.Companion.EXPIRED: MediaKeyStatus get() = "expired".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyStatus.Companion.RELEASED: MediaKeyStatus get() = "released".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyStatus.Companion.OUTPUT_RESTRICTED: MediaKeyStatus get() = "output-restricted".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyStatus.Companion.OUTPUT_DOWNSCALED: MediaKeyStatus get() = "output-downscaled".asDynamic().unsafeCast<MediaKeyStatus>()
public inline val MediaKeyStatus.Companion.STATUS_PENDING: MediaKeyStatus get() = "status-pending".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyStatus.Companion.INTERNAL_ERROR: MediaKeyStatus get() = "internal-error".asDynamic().unsafeCast<MediaKeyStatus>()

public inline val MediaKeyMessageTypeDef.Companion.LICENSE_REQUEST: MediaKeyMessageType get() = "license-request".asDynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageTypeDef.Companion.LICENSE_RENEWAL: MediaKeyMessageType get() = "license-renewal".asDynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageTypeDef.Companion.LICENSE_RELEASE: MediaKeyMessageType get() = "license-release".asDynamic().unsafeCast<MediaKeyMessageType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaKeyMessageType {
    companion object
}

public inline val MediaKeyMessageType.Companion.LICENSE_REQUEST: MediaKeyMessageType get() = "license-request".asDynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageType.Companion.LICENSE_RENEWAL: MediaKeyMessageType get() = "license-renewal".asDynamic().unsafeCast<MediaKeyMessageType>()

public inline val MediaKeyMessageType.Companion.LICENSE_RELEASE: MediaKeyMessageType get() = "license-release".asDynamic().unsafeCast<MediaKeyMessageType>()
EventTarget?

open val region: String?
open val pageX: Double
open val pageY: Double
open val pageOffsetX: Double
open val pageOffsetY: Double

fun getModifierState(keyArg: String?): Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

public external interface MouseEventInit : EventModifierInit {
    var screenX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var screenY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var clientX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var clientY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var button: Short? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var buttons: Short? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var region: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MouseEventInit(screenX: Int? = 0, screenY: Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? /* = null */
    
    var ctrlKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierAltGraph: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierCapsLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierFn: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierFnLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierHyper: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierNumLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierScrollLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierSuper: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierSymbol: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierSymbolLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var view: Window? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var detail: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var bubbles: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var composed: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var returnOrigin: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["returnOrigin"] = returnOrigin
}

Open Source Used in DNAC Platform Arica 8170
    val o = js("({})")
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```javascript
js(\"([\ ]*)\")\n  o["data"] = data\n  o["isComposing"] = isComposing\n  o["view"] = view\n  o["detail"] = detail\n  o["bubbles"] = bubbles\n  o["cancelable"] = cancelable\n  o["composed"] = composed\n
* Exposes the JavaScript

[KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin
*/\n
open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {
  open var key: String? = null
  open var code: String? = null
  open var location: Int? = 0
  open var ctrlKey: Boolean? = null
  open var shiftKey: Boolean? = null
  open var altKey: Boolean? = null
  open var metaKey: Boolean? = null
  open var repeat: Boolean? = null
  open var isComposing: Boolean? = null
  open var charCode: Int? = null
  open var keyCode: Int? = null
  open var which: Int? = null
  fun getModifierState(keyArg: String): Boolean

  companion object {
    val DOM_KEY_LOCATION_STANDARD: Int
    val DOM_KEY_LOCATION_LEFT: Int
    val DOM_KEY_LOCATION_RIGHT: Int
    val DOM_KEY_LOCATION_NUMPAD: Int
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
  }
}

public interface KeyboardEventInit : EventModifierInit {
  var key: String? /* = "" */
  var code: String? /* = "" */
  var location: Int? /* = 0 */
  var repeat: Boolean? /* = false */
  var isComposing: Boolean? /* = false */
  var ctrlKey: Boolean? /* = false */
  var shiftKey: Boolean? /* = false */
  var altKey: Boolean? /* = false */
  var metaKey: Boolean? /* = false */
  var modifierAltGraph: Boolean? /* = false */
  var modifierCapsLock: Boolean? /* = false */
  var modifierFn: Boolean? /* = false */
  var modifierHyper: Boolean? /* = false */
  var modifierNumLock: Boolean? /* = false */
  var modifierScrollLock: Boolean? /* = false */
  var modifierSuper: Boolean? /* = false */
  var modifierSymbol: Boolean? /* = false */
  var modifierSymbolLock: Boolean? /* = false */
  var view: Window? = null
  var detail: Int? = 0
  var bubbles: Boolean? = false
  var cancelable: Boolean? = false
  var composed: Boolean? = false

  @kotlin.internal.InlineOnly
    isComposing: Boolean? = false, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false,
    metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false,
    modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false,
    KeyboardEventInit {
      val o = js("({})")
      o["key"] = key
      o["code"] = code
      o["location"] = location
      o["repeat"] = repeat
      o["isComposing"] = isComposing
      o["ctrlKey"] = ctrlKey
      o["shiftKey"] = shiftKey
      o["altKey"] = altKey
      o["metaKey"] = metaKey
      o["modifierAltGraph"] = modifierAltGraph
      o["modifierCapsLock"] = modifierCapsLock
      o["modifierFn"] = modifierFn
      o["modifierHyper"] = modifierHyper
      o["modifierNumLock"] = modifierNumLock
      o["modifierScrollLock"] = modifierScrollLock
      o["modifierSuper"] = modifierSuper
      o["modifierSymbol"] = modifierSymbol
      o["modifierSymbolLock"] = modifierSymbolLock
      o["view"] = view
      o["detail"] = detail
      o["bubbles"] = bubbles
      o["cancelable"] = cancelable
      o["composed"] = composed
      return o
    }
}

/**
 * Exposes the JavaScript
[CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin
 */
open class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) : UIEvent {
  open var data: String? = null
  open var view: Window? = null
  open var detail: Int? = 0
  open var bubbles: Boolean? = false
  open var cancelable: Boolean? = false
  open var composed: Boolean? = false

  @kotlin.internal.InlineOnly
  public inline fun CompositionEventInit(data: String? = "", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false, composed: Boolean? = false): CompositionEventInit {
    val o = js(\"([\ ]*)\")
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
  }
}

/**
 * Exposes the JavaScript
[Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin
 */
open class Event(type: String, eventInitDict: EventInit = definedExternally) {
  open var type: String? = null
  open var currentTarget: EventTarget? = null
  open var defaultPrevented: Boolean? = false

  @kotlin.internal.InlineOnly
    EventInit {
      val o = js(\"([\ ]*)\")
      o["type"] = type
      o["currentTarget"] = currentTarget
      o["defaultPrevented"] = defaultPrevented
      return o
    }
}
```

Open Source Used In DNAC Platform Arica 8172
Boolean open val isTrusted: Boolean open val timeStamp: Number fun composedPath():
Array<EventTarget> fun stopPropagation() fun stopImmediatePropagation() fun preventDefault()
fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean) companion object { Open Source Used In DNAC Platform Arica

Boolean
open val isTrusted: Boolean
open val timeStamp: Number
fun composedPath():
Array<EventTarget>
fun stopPropagation()
fun stopImmediatePropagation()
fun preventDefault()
fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)

Boolean
open val isTrusted: Boolean
open val timeStamp: Number
fun composedPath():
Array<EventTarget>
fun stopPropagation()
fun stopImmediatePropagation()
fun preventDefault()
fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)

Boolean
open val isTrusted: Boolean
open val timeStamp: Number
fun composedPath():
Array<EventTarget>
fun stopPropagation()
fun stopImmediatePropagation()
fun preventDefault()
fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)

Boolean
open val isTrusted: Boolean
open val timeStamp: Number
fun composedPath():
Array<EventTarget>
fun stopPropagation()
fun stopImmediatePropagation()
fun preventDefault()
fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)
/**
 * Exposes the JavaScript [HTMLUnknownElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUnknownElement) to Kotlin
 */
public external abstract class HTMLUnknownElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin
 */
public external abstract class HTMLHtmlElement : HTMLElement {
    open var version: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin
 */
public external abstract class HTMLHeadElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTitleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTitleElement) to Kotlin
 */
public external abstract class HTMLTitleElement : HTMLElement {
    open var text: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Open Source Used In DNAC Platform Arica 8175

CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLBaseElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBaseElement) to Kotlin */
public external abstract class HTMLBaseElement : HTMLElement {
    open var href: String
    open var target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin */
public external abstract class HTMLLinkElement : HTMLElement, LinkStyle {
    open var href: String
    open var crossOrigin: String?
    open var rel: String
    open var `as`: RequestDestination
    open val relList: DOMTokenList
    open var media: String
    open var nonce: String
    open var hreflang: String
    open var type: String
    open val sizes: DOMTokenList
    open var referrerPolicy: String
    open var charset: String
    open var rev: String
    open var target: String
    open var scope: String
    open var workerType: WorkerType

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin */
public external abstract class HTMLMetaElement : HTMLElement {
    open var name: String
    open var httpEquiv: String
    open var content: String
    open var scheme: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin */
public
external abstract class HTMLStyleElement : HTMLElement, LinkStyle {
    open var media: String
    open var nonce: String
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin
 */
public external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {
    open var text: String
    open var link: String
    open var vLink: String
    open var aLink: String
    open var bgColor: String
    open var background: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin
 */
public external abstract class HTMLHeadingElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin
 */
public external abstract class HTMLParagraphElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin
 */
public external abstract class HTMLHRElement : HTMLElement {
    open var align: String
    open var color: String
}
Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLIElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLIElement) to Kotlin
 */
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDListElement) to Kotlin
 */
public external abstract class HTMLDListElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin
 */
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin
 */
public external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var hreflang: String
    open var type: String
    open var text: String
    open var referrerPolicy: String
    open var coords: String
    open var charset: String
    open var name: String
    open var rev: String
    open var shape: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
The document contains code snippets related to JavaScript classes and their attributes, such as `HTMLDataElement`, `HTMLTimeElement`, `HTMLSpanElement`, and `HTMLBRElement`. These classes are used to expose JavaScript functionality to Kotlin. Each class has a `value` property of type `String`, and some also have other properties like `dateTime` and `clear`. The code is structured with import statements and class definitions, followed by the class implementations with properties and companion objects.
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

// Exposes the JavaScript [HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to Kotlin

// *public* external interface HTMLHyperlinkElementUtils

var href: String
val origin: String
var protocol: String
var username: String
var password: String
var host: String
var hostname: String
var port: String
var pathname: String
var search: String
var hash: String

/**
 * Exposes the JavaScript [HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin
 */
public external abstract class HTMLModElement : HTMLElement {
open var cite: String
open var dateTime: String

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

// Exposes the JavaScript [HTMLPictureElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement) to Kotlin

// *public* external abstract class HTMLPictureElement : HTMLElement {

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

// Exposes the JavaScript [HTMLSourceElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement) to Kotlin

// *public* external abstract class HTMLSourceElement : HTMLElement {

open var src: String
open var type: String
open var srcset: String
open var sizes: String
open var media: String

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

// Exposes the JavaScript [HTMLImageElement](https://developer.mozilla.org/en/docs/Web/API/HTMLImageElement) to Kotlin

// *public* external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {

open var alt: String
open var src: String
open var srcset: String
open var sizes: String
open var crossOrigin: String?
open var useMap: String
open var isMap: Boolean
open var width: Int
open var height: Int
open var naturalWidth: Int
open var naturalHeight: Int
open var complete: Boolean
open var currentSrc: String
open var referrerPolicy: String
open var name: String
open var lowsrc: String;
    open var align: String;
    open var hspace: Int;
    open var vspace: Int;
    open var longDesc: String;
    open var border: String;
    open val x: Int;
    open val y: Int;

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin
 */
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String;
    open var srcdoc: String;
    open var name: String;
    open val sandbox: DOMTokenList;
    open var allowFullscreen: Boolean;
    open var allowUserMedia: Boolean;
    open var width: String;
    open var height: String;
    open var referrerPolicy: String;
    open var contentDocument: Document;
    open var contentWindow: Window;
    open var align: String;
    open var scrolling: String;
    open var frameborder: String;
    open var longDesc: String;
    open var marginHeight: String;
    open var marginWidth: String;
    fun getSVGDocument(): Document;

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin
 */
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String;
    open var type: String;
    open var width: String;
    open var height: String;
    open var align: String;
    open var name: String;
    fun getSVGDocument(): Document;

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin
 */
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String;
    open var type: String;
    open var typeMustMatch: Boolean;
    open var name: String;
    open var useMap: String;
    open val form: HTMLFormElement;
    open var width: String;
    open var height: String;
    open val contentDocument: Document;
    open val contentWindow: Window;
    open val willValidate: Boolean;
    open val validity: ValidityState;
    open val validationMessage: String;
    open var align: String;
    open var archive: String;
    open var code: String;
    open var declare: Boolean;
    open var hspace: Int;
    open var standby: String;
    open var vspace: Int;
    open var codeBase: String;
    open var codeType: String;
    open var open var border:
String
    fun getSVGDocument(): Document?
    fun checkValidity(): Boolean
    fun reportValidity():

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript
 [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin
 */
public external abstract class HTMLParamElement : HTMLElement {
    open var name: String
    open var value: String
    open var type: String
    open var valueType: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript
 [HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin
 */
public external abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    open var videoWidth: Int
    open var videoHeight: Int
    open var poster: String
    open var playsInline: Boolean

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript
 [HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin
 */
public external abstract class HTMLAudioElement : HTMLMediaElement {
    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

/**
 * Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin
 */
public external abstract class HTMLTrackElement : HTMLElement {
    open var kind: String
    open var src: String
    open var srclang: String
    open var label: String
    open var default: Boolean
    open val readyState: Short
    open val track: TextTrack

    companion object {
        val NONE: Short
        val LOADING: Short
        val LOADED: Short
        val ERROR: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to Kotlin
 */
public external abstract class HTMLMediaElement : HTMLElement {
    open val error: MediaError?
    open var src: String
    open var srcObject: MediaProvider?
    open val currentSrc: String
    open var crossOrigin: String?
    open val networkState: Short
    open var preload: String
    open val buffered: TimeRanges
    open var readyState: Short
    open var seeking: Boolean
    open var currentTime: Double
    open val duration: Double
    open var paused: Boolean
    open var defaultPlaybackRate: Double
    open var playbackRate: Double
    open var audioTracks: AudioTrackList
    open var videoTracks: VideoTrackList
    open var textTracks: TextTrackList
    open var mediaKeys: MediaKeys?
    open var onencrypted: ((Event) -> dynamic)?
    open var onwaitingforkey: ((Event) -> dynamic)?
    fun load()
    fun canPlayType(type: String): CanPlayTypeResult
    fun fastSeek(time: Double)
    fun getStartDate(): dynamic
    fun play(): Promise<Unit>
    fun pause()
    fun addTextTrack(kind: TextTrackKind, label: String = definedExternally, language: String = definedExternally): TextTrack
    fun setMediaKeys(mediaKeys: MediaKeys?): Promise<Unit>

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin
 */
public external abstract class MediaError {
    open val code: Short

    companion object {
        val MEDIA_ERR_ABORTED: Short
    }
}
val MEDIA_ERR_NETWORK: Short
val MEDIA_ERR-carousel: Short
val MEDIA_ERR_SRC_NOT_SUPPORTED: Short

[AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin

```kotlin
open class AudioTrackList : EventTarget {
    open val length: Int
    open var onchange: (Event) -> dynamic
    open var onaddtrack: ((TrackEvent) -> dynamic)
    fun getTrackByIndex(id: String): AudioTrack

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}
```

```kotlin
open class AudioTrack : UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var enabled: Boolean
    open val sourceBuffer: SourceBuffer?
}
```

```kotlin
open external abstract class VideoTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)
    open var onaddtrack: ((TrackEvent) -> dynamic)
    fun getTrackByIndex(id: String): VideoTrack
}
```

```kotlin
open class VideoTrack : UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var selected: Boolean
    open val sourceBuffer: SourceBuffer?
}
```

```kotlin
open external abstract class TextTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)
    open var onaddtrack: ((TrackEvent) -> dynamic)
    fun getTrackByIndex(id: String): TextTrack
}
```

```kotlin
open class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val kind: TextTrackKind
    open val label: String
    open val language: String
    open val id: String
    open val inBandMetadataDispatchType: String
    open var mode: TextTrackMode
    open val cues: TextTrackCueList?
    open val activeCues: TextTrackCueList?
    open var oncuechange: ((Event) -> dynamic)
    fun addCue(cue: TextTrackCue)
    fun removeCue(cue: TextTrackCue)
}
```

```kotlin
open external abstract class TextTrackCueList {
    open val length: Int
    fun getCueById(id: String): TextTrackCue?
}
```

```kotlin
open class TextTrackCue : EventTarget {
    open val track: TextTrack?
    open var id: String
    open var startTime: Double
    open var endTime: Double
    open var pauseOnExit: Boolean
    open var onenter: ((Event) -> dynamic)
    open var onexit: ((Event) -> dynamic)
}
```

```kotlin
open class TimeRanges {
    open val length: Int
    fun start(index: Int): Double
    fun end(index: Int): Double
}
```

```kotlin
open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {
    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?
    fun getTrackByIndex(id: String): AudioTrack
}
```

```kotlin
open interface TrackEventInit : EventInit {
    var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
open fun TrackEventInit(track:
    val o = js("({})")
    o["track"] = track
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin
 */
public external abstract class HTMLMapElement : HTMLElement {
    open var name: String
    open val areas: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin
 */
public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var alt: String
    open var coords: String
    open var shape: String
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var referrerPolicy: String
    open var noHref: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var caption: HTMLTableCaptionElement?
    open var tHead: HTMLTableSectionElement?
    open var tFoot: HTMLTableSectionElement?
    open var tBodies: HTMLTableCaptionElement?
    open var tRows: HTMLCollection
    open var align: String
    open var border: String
    open var frame: String
    open var rules: String
    open var summary: String
    open var width: String
    open var bgColor: String
    open var cellPadding: String
    open var cellSpacing: String
    fun createCaption(): HTMLTableCaptionElement
    fun deleteCaption()
    fun createTHead(): HTMLTableCaptionElement
    fun deleteTHead()
    fun createTBody(): HTMLTableCaptionElement
    fun deleteTBody(): HTMLTableCaptionElement
    fun insertRow(index: Int = definedExternally): HTMLTableRowElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
Open Source Used In DNAC Platform Arica 8186
/**
 * Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin
 */
public external abstract class HTMLTableCellElement : HTMLElement {
    open var colSpan: Int
    open var rowSpan: Int
    open var headers: String
    open var cellIndex: Int
    open var scope: String
    open var abbr: String
    open var align: String
    open var axis: String
    open var height: String
    open var width: String
    open var ch: String
    open var chOff: String
    open var noWrap: Boolean
    open var vAlign: String
    open var bgColor: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin
 */
public external abstract class HTMLFormElement : HTMLElement {
    open var acceptCharset: String
    open var action: String
    open var autocomplete: String
    open var enctype: String
    open var encoding: String
    open var method: String
    open var name: String
    open var noValidate: Boolean
    open var target: String
    open var elements: HTMLFormControlsCollection
    open var length: Int
    fun submit()
    fun reset()
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin
 */
public external abstract class HTMLLabelElement : HTMLElement {
    open var htmlFor: String
    open var control: HTMLElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin
 */
public
external abstract class HTMLInputElement : HTMLElement {
    var open var accept: String
    var open var autofocus: Boolean
    var open var formAction: String
    var open var formMethod: String
    var open var formNoValidate: Boolean
    var open var formTarget: String
    var open var formValid: Boolean
    var open var inputMode: String
    var open var list: HTMLFormElement
    var open var max: String
    var open var maxLen: Int
    var open var min: String
    var open var minLength: Int
    var open var multiple: Boolean
    var open var name: String
    var open var placeholder: String
    var open var readOnly: Boolean
    var open var required: Boolean
    var open var size: Int
    var open var type: String
    var open var value: String
    var open var valueAsDate: dynamic
    var open var valueAsNumber: Double
    var open var width: Int
    var open var willValidate: Boolean
    var open var validity: ValidityState
    var open var validationMessage: String
    val labels: NodeList
    fun stepUp(n: Int = definedExternally)
    fun stepDown(n: Int = definedExternally)
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    fun select()
    fun setRangeText(replacement: String)
    fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
    fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin
 */
public external abstract class HTMLButtonElement : HTMLElement {
    var autofocus: Boolean
    var disabled: Boolean
    var form: HTMLFormElement
    var formAction: String
    var formEnctype: String
    var formMethod: String
    var formNoValidate: Boolean
    var formTarget: String
    var name: String
    var type: String
    var value: String
    var menu: HTMLMenuElement
    var willValidate: Boolean
    var validity: ValidityState
    var validationMessage: String
    val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin
 */
public external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {
    var autocomplete: String
    var autofocus: Boolean
    var disabled: Boolean
    var form: HTMLFormElement
    var formAction: String
    var formMethod: String
    var formNoValidate: Boolean
    var formTarget: String
    var name: String
    var multiple: Boolean
    var value: String
    var items: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Boolean
    open var value: String
    open var text: String
    open val index: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 * [HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement) to Kotlin
 *
 * @open
 * @var value: String
 * @var text: String
 * @open val index: Int
 *
 * @companion object {
 *     val ELEMENT_NODE: Short
 *     val ATTRIBUTE_NODE: Short
 *     val TEXT_NODE: Short
 *     val CDATA_SECTION_NODE: Short
 *     val ENTITY_REFERENCE_NODE: Short
 *     val ENTITY_NODE: Short
 *     val PROCESSING_INSTRUCTION_NODE: Short
 *     val COMMENT_NODE: Short
 *     val DOCUMENT_NODE: Short
 *     val DOCUMENT_TYPE_NODE: Short
 *     val DOCUMENT_FRAGMENT_NODE: Short
 *     val NOTATION_NODE: Short
 *     val DOCUMENT_POSITION_DISCONNECTED: Short
 *     val DOCUMENT_POSITION_PRECEDING: Short
 *     val DOCUMENT_POSITION_FOLLOWING: Short
 *     val DOCUMENT_POSITION_CONTAINS: Short
 *     val DOCUMENT_POSITION_CONTAINED_BY: Short
 * }
 */
public external abstract class HTMLTextAreaElement : HTMLElement {
    open var autocomplete: String
    open var autofocus: Boolean
    open var cols: Int
    open var dirName: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var inputMode: String
    open var maxLength: Int
    open var minLength: Int
    open var name: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var rows: Int
    open var wrap: String
    open val type: String
    open var defaultValue: String
    open var value: String
    open val textLength: Int
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    open var selectionStart: Int?
    open var selectionEnd: Int?
    open var selectionDirection: String?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    fun select()
    fun setRangeText(replacement: String)
    fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
    fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 * [HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement) to Kotlin
 *
 * @open
 * @var value: String
 * @var text: String
 * @open val index: Int
 *
 * @companion object {
 *     val ELEMENT_NODE: Short
 *     val ATTRIBUTE_NODE: Short
 *     val TEXT_NODE: Short
 *     val CDATA_SECTION_NODE: Short
 *     val ENTITY_REFERENCE_NODE: Short
 *     val ENTITY_NODE: Short
 *     val PROCESSING_INSTRUCTION_NODE: Short
 *     val COMMENT_NODE: Short
 *     val DOCUMENT_NODE: Short
 *     val DOCUMENT_TYPE_NODE: Short
 *     val DOCUMENT_FRAGMENT_NODE: Short
 *     val NOTATION_NODE: Short
 *     val DOCUMENT_POSITION_DISCONNECTED: Short
 *     val DOCUMENT_POSITION_PRECEDING: Short
 *     val DOCUMENT_POSITION_FOLLOWING: Short
 *     val DOCUMENT_POSITION_CONTAINS: Short
 *     val DOCUMENT_POSITION_CONTAINED_BY: Short
 * }
 */
public external abstract class HTMLKeygenElement : HTMLElement {
    open var autofocus: Boolean
    open var challenge: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var keytype: String
    open var name: String
    open val type: String
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 * [HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin
 *
 * @open
 * @var value: String
 * @var text: String
 * @open val index: Int
 *
 * @companion object {
 *     val ELEMENT_NODE: Short
 *     val ATTRIBUTE_NODE: Short
 *     val TEXT_NODE: Short
 *     val CDATA_SECTION_NODE: Short
 *     val ENTITY_REFERENCE_NODE: Short
 *     val ENTITY_NODE: Short
 *     val PROCESSING_INSTRUCTION_NODE: Short
 *     val COMMENT_NODE: Short
 *     val DOCUMENT_NODE: Short
 *     val DOCUMENT_TYPE_NODE: Short
 *     val DOCUMENT_FRAGMENT_NODE: Short
 *     val NOTATION_NODE: Short
 *     val DOCUMENT_POSITION_DISCONNECTED: Short
 *     val DOCUMENT_POSITION_PRECEDING: Short
 *     val DOCUMENT_POSITION_FOLLOWING: Short
 *     val DOCUMENT_POSITION_CONTAINS: Short
 *     val DOCUMENT_POSITION_CONTAINED_BY: Short
 * }
 */
```kotlin
/*\n**\n*/
public external abstract class HTMLOutputElement : HTMLElement {
    open val htmlFor: DOMTokenList
    open val form: HTMLFormElement?
    open var name: String
    open var type: String
    open var defaultValue: String
    open var value: String
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin
*/
public external abstract class HTMLProgressElement : HTMLElement {
    open var value: Double
    open var max: Double
    open val position: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin
*/
public external abstract class HTMLMeterElement : HTMLElement {
    open var value: Double
    open var min: Double
    open var max: Double
    open var low: Double
    open var high: Double
    open var optimum: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin
*/
public external abstract class HTMLFieldSetElement : HTMLElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var name: String
    open val type: String
    open val elements: HTMLCollection
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```
radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean =
definedExternally

CanvasGradient [https://developer.mozilla.org/en/docs/Web/API/CanvasGradient] to Kotlin

CanvasPattern [https://developer.mozilla.org/en/docs/Web/API/CanvasPattern] to Kotlin

CanvasPattern [https://developer.mozilla.org/en/docs/Web/API/TextMetrics] to Kotlin

SymbolPattern [https://developer.mozilla.org/en/docs/Web/API/TextMetrics] to Kotlin

HTMLCanvasElement to Kotlin


ImageData [https://developer.mozilla.org/en/docs/Web/API/ImageData] to Kotlin

WindowOrWorkerGlobalScope, WindowSessionStorage, WindowLocalStorage, GlobalPerformance,
UnionMessagePortOrWindowProxy { override val window: Window
    override val self: Window
    override val document: Document
    override var name: String
    override val location: Location
    override val history: History
    override open val customElements: CustomElementRegistry
    override open val navigator: Navigator
    override val applicationCache: ApplicationCache
    override val external: External
    override val screen: Screen
    override val innerWidth: Int
    override val innerHeight: Int
    override var opener: Any?
    override val parent: Window
    override val frameElement: Element?
    override val navigator: Navigator
    override val applicationCache: ApplicationCache
    override val external: External
    override val screen: Screen
    override val innerWidth: Int
    override val innerHeight: Int
    override val devicePixelRatio: Double
    fun close()
    fun stop()
    fun focus()
    fun blur()
    fun open(url: String = definedExternally, target: String = definedExternally, features: String = definedExternally): Window?
    fun alert()
    fun alert(message: String)
    fun confirm(message: String = definedExternally): Boolean
    fun prompt(message: String = definedExternally, default: String = definedExternally): String?
    fun print()
    fun requestAnimationFrame(callback: (Double) -> Unit): Int
    fun cancelAnimationFrame(handle: Int)
    fun postMessage(message: Any?, targetOrigin: String, transfer: Array<dynamic> = definedExternally)
    fun captureEvents()
    fun releaseEvents()
    fun matchMedia(query: String): MediaQueryList
    fun moveTo(x: Int, y: Int)
    fun moveBy(x: Int, y: Int)
    fun resizeTo(x: Int, y: Int)
    fun resizeBy(x: Int, y: Int)
    fun scroll(options: ScrollToOptions = definedExternally)
    fun scroll(x: Double, y: Double)
    fun scrollTo(options: ScrollToOptions = definedExternally)
    fun scrollBy(options: ScrollToOptions = definedExternally)
    fun getComputedStyle(elt: Element, pseudoElt: String? = definedExternally): CSSStyleDeclaration
}

@kotlin.internal.InlineOnly
public inline operator fun Window.get(name: String): dynamic = asDynamic()[name]

public external abstract class BarProp {
    open val visible: Boolean
}

/**
 * Exposes the JavaScript
[
[History](https://developer.mozilla.org/en/docs/Web/API/History) to Kotlin
*/
public external abstract class History {
    open val length: Int
    open var scrollRestoration: ScrollRestoration
    open val state: Any?
    fun go(delta: Int = definedExternally)
    fun back()
    fun forward()
    fun pushState(data: Any?, title: String?, url: String? = definedExternally)
    fun replaceState(data: Any?, title: String?, url: String? = definedExternally)
}

/**
 * Exposes the JavaScript
[
[Location](https://developer.mozilla.org/en/docs/Web/API/Location) to Kotlin
*/
public external open class Location {
    open var href: String
    open val origin: String
    open val protocol: String
    open val host: String
    open val hostname: String
    open val port: String
    open val pathname: String
    open val search: String
    open val hash: String
    open val ancestorOrigins: Array<out String>
    fun assign(url: String)
    fun replace(url: String)
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["state"] = state
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[
[HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin
*/
public open class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) : Event {
    fun get(targetId: String)
}

/**
 * Exposes the JavaScript
[
[EventTarget](https://developer.mozilla.org/en/docs/Web/API/EventTarget) to Kotlin
*/
public open class EventTarget {
    fun addEventListener(target: String)
    fun removeEventListener(target: String)
    fun dispatchEvent(target: String)
}

/**
 * Exposes the JavaScript
[
[Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin
*/
public external class Window {
    open val window: Window
    open val self: Window
    open val document: Document
    open var name: String
    open val location: Location
    open val history: History
    open val customElements: CustomElementRegistry
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val innerWidth: Int
    open val innerHeight: Int
    open val devicePixelRatio: Double
    fun close()
    fun stop()
    fun focus()
    fun blur()
    fun open(url: String = definedExternally, target: String = definedExternally, features: String = definedExternally): Window?
    fun alert()
    fun alert(message: String)
    fun confirm(message: String = definedExternally): Boolean
    fun prompt(message: String = definedExternally, default: String = definedExternally): String?
    fun print()
    fun requestAnimationFrame(callback: (Double) -> Unit): Int
    fun cancelAnimationFrame(handle: Int)
    fun postMessage(message: Any?, targetOrigin: String, transfer: Array<dynamic> = definedExternally)
    fun captureEvents()
    fun releaseEvents()
    fun matchMedia(query: String): MediaQueryList
    fun moveTo(x: Int, y: Int)
    fun moveBy(x: Int, y: Int)
    fun resizeTo(x: Int, y: Int)
    fun resizeBy(x: Int, y: Int)
    fun scroll(options: ScrollToOptions = definedExternally)
    fun scroll(x: Double, y: Double)
    fun scrollTo(options: ScrollToOptions = definedExternally)
    fun scrollBy(options: ScrollToOptions = definedExternally)
    fun getComputedStyle(elt: Element, pseudoElt: String? = definedExternally): CSSStyleDeclaration
}

@kotlin.internal.InlineOnly
public inline operator fun Window.get(name: String): dynamic = asDynamic()[name]
Short
|\n|\n\n
public external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\n}\n@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}

@\n\n\n\npublic external interface HashChangeEventInit : EventInit {\n\nvar oldURL: String? /* = "" */\nget() = definedExternally\nset(value) = definedExternally\nvar newURL: String? /* = "" */\n\nget() = definedExternally\nset(value) = definedExternally\n}
dynamic):?
    get() = definedExternally
    set(value) = definedExternally
    var onmousemove:
        (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseout: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseover: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmouseup: (MouseEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onwheel: (WheelEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpause: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onplay: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onplaying: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onprogress: (ProgressEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onratechange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onreset: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onresize: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onscroll: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onseeked: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onseeking: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onselect: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onshow: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onstalled: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onsubmit: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onsuspend: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ontimeupdate: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ontoggle: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onvolumechange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onwaiting: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ongotpointercapture: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onlostpointercapture: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerdown: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointermove: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerup: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointercancel: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerover: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerout: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerenter: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpointerleave: (PointerEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onbeforeprint: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onbeforeunload: (BeforeUnloadEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onhashchange: (HashChangeEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onlanguagechange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onmessage: (MessageEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onoffline: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ononline: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpagehide: (PageTransitionEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpageshow: (PageTransitionEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpagemenu: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpopstate: (PopStateEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onorientationchange: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpopin: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var onpopstate: (Event) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ontransitionend: (TransitionEvent) -> dynamic?
        get() = definedExternally
        set(value) = definedExternally
    var ontransitionstart: (TransitionEvent) -> dynamic?
set(value) = definedExternally
    var onpageshow: ((PageTransitionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpopstate: ((PopStateEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onrejectionhandled: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onstorage: ((StorageEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onunload: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

public external interface DocumentAndElementEventHandlers {
    var oncopy: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncut: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpaste: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
 */
public external interface WindowOrWorkerGlobalScope {
    val origin: String
    val caches: CacheStorage
    fun btoa(data: String): String
    fun atob(data: String): String
    fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearTimeout(handle: Int = definedExternally)
    fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearInterval(handle: Int = definedExternally)
    fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>
    fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>
}

/**
 * Exposes the JavaScript
 */
public external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {
    open val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val maxTouchPoints: Int
    open val serviceWorker: ServiceWorkerContainer
    fun requestMediaKeySystemAccess(keySystem: String, supportedConfigurations: Array<MediaKeySystemConfiguration>): Promise<MediaKeySystemAccess>
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val platform: String
    val product: String
    val productSub: String
    val userAgent: String
    val vendor: String
    val vendorSub: String
    fun taintEnabled(): Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorLanguage {
    val language: String
    val languages: Array<out String>
}

public external interface NavigatorContentUtils {
    fun registerProtocolHandler(scheme: String, url: String, title: String)
    fun registerContentHandler(mimeType: String, url: String, title: String)
    fun isProtocolHandlerRegistered(scheme: String, url: String): String
    fun isContentHandlerRegistered(mimeType: String, url: String): String
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorPlugins {
    val plugins: PluginArray
    val mimeTypes: MimeTypeArray
    fun javaEnabled(): Boolean
}

@kotlin.internal.InlineOnly
public
inline operator fun PluginArray.get(index: Int): Plugin? =
asDynamic()[index]\n@kotlin.internal.InlineOnly\npublic inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]\n/**
 * Exposes the JavaScript
 */
[MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin\n**\n/public external abstract class MimeTypeArray : ItemArrayLike<MimeType> {\n    override fun item(index: Int): MimeType? = asDynamic()[index]\n/**
 * Exposes the JavaScript
 */
[ImageBitmap](https://developer.mozilla.org/en/docs/Web/API/ImageBitmap) to Kotlin\n**\n/public external abstract class ImageBitmap : CanvasImageSource, TexImageSource {\n    override fun close() = definedExternally
}
namespace:

public external class ImageBitmapOptions {
    var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */
    var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */
    var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */
    var resizeWidth: Int?
    var resizeHeight: Int?
    var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */
}
namespace:

public external class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {\n    val data: Any?
    var origin: String
    var lastEventId: String
    var source: UnionMessagePortOrWindowProxy?
    var ports: Array<MessagePort>
}
namespace:

public external interface MessageEventInit : EventInit {
    var data: Any
    var origin: String
    var lastEventId: String
    var source: UnionMessagePortOrWindowProxy?
    var ports: Array<MessagePort>
}

definedExternally
} @ kotlin.internal.InlineOnly
    return o
}["data"] = data
["origin"] = origin
["lastEventId"] = lastEventId
["source"] = source
["ports"] = ports
["bubbles"] = bubbles
["cancelable"] = cancelable
["composed"] = composed
return o
} */ * Exposes the JavaScript
[EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin */ *unpublic external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
    withCredentials: Boolean? = false */ * get() = definedExternally
} @ kotlin.internal.InlineOnly
public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
    return o
} */ * Exposes the JavaScript
[WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin */ *unpublic external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
    return o
} */ * Exposes the JavaScript
[CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin */ *unpublic external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {
    return o
} */ * Exposes the JavaScript
[MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin */ *unpublic external open class MessageChannel {
    return o
} */ * Exposes the JavaScript
[BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin */ *unpublic external abstract class BroadcastChannel(name: String) : EventTarget, UnionMessagePortOrWindowProxy, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    return o
} */ * Exposes the JavaScript
[WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin */ *unpublic external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope, GlobalPerformance {
    return o
}
navigator: WorkerNavigator
    open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    open var onlanguagechange: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun close(): Unit

** Exposes the JavaScript

[DedicatedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/DedicatedWorkerGlobalScope) to Kotlin

```kotlin
@public external abstract class DedicatedWorkerGlobalScope : WorkerGlobalScope {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun close()
}
```

** Exposes the JavaScript

[SharedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/SharedWorkerGlobalScope) to Kotlin

```kotlin
@public external abstract class SharedWorkerGlobalScope : WorkerGlobalScope {
    open val name: String
    open val applicationCache: ApplicationCache
    open var onconnect: ((Event) -> dynamic)?
    fun close()
}
```

** Exposes the JavaScript

[AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin

```kotlin
@public external interface AbstractWorker {
    var onerror: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}
```

** Exposes the JavaScript

[Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin

```kotlin
@public external open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    var onmessage: ((MessageEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    fun terminate()
}
```

** Exposes the JavaScript

[WorkerOptions](https://developer.mozilla.org/en/docs/Web/API/WorkerOptions) to Kotlin

```kotlin
@public external interface WorkerOptions {
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
    var credentials: RequestCredentials? /* = RequestCredentials.OMIT */
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun WorkerOptions(type: WorkerType? = WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {
    val o = js("({})")
    o["type"] = type
    o["credentials"] = credentials
    return o
}
```

** Exposes the JavaScript

[SharedWorker](https://developer.mozilla.org/en/docs/Web/API/SharedWorker) to Kotlin

```kotlin
@public external open class SharedWorker(scriptURL: String, name: String = definedExternally, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    open val port: MessagePort
    override var onerror: ((Event) -> dynamic)?
}
```

** Exposes the JavaScript

[NavigatorConcurrentHardware](https://developer.mozilla.org/en/docs/Web/API/NavigatorConcurrentHardware) to Kotlin

```kotlin
@public external interface NavigatorConcurrentHardware {
    val hardwareConcurrency: Number
}
```

** Exposes the JavaScript

[WorkerNavigator](https://developer.mozilla.org/en/docs/Web/API/WorkerNavigator) to Kotlin

```kotlin
@public external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorConcurrentHardware {
    var serviceWorker: ServiceWorkerContainer
}
```

** Exposes the JavaScript

[WorkerLocation](https://developer.mozilla.org/en/docs/Web/API/WorkerLocation) to Kotlin

```kotlin
@public external abstract class WorkerLocation {
    open val href: String
    open val origin: String
    open val protocol: String
    open val hostname: String
    open val port: String
    open val pathname: String
    open val search: String
    open val hash: String
}
```

** Exposes the JavaScript

[Storage](https://developer.mozilla.org/en/docs/Web/API/Storage) to Kotlin

```kotlin
@public external abstract class Storage {
    open val length: Int
    fun key(index: Int): String?
    fun removeItem(key: String)
    fun clear()
    fun getItem(key: String): String?
    fun setItem(key: String, value: String)
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun Storage.get(key: String): String? = asDynamic()[key]
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun Storage.set(key: String, value: String) { asDynamic()[key] = value }
```

** Exposes the JavaScript

[WindowSessionStorage](https://developer.mozilla.org/en/docs/Web/API/WindowSessionStorage) to Kotlin

```kotlin
@public external interface WindowSessionStorage {
    val sessionStorage: Storage
}
```

** Exposes the JavaScript

[Open Source Used In DNAC Platform Arica 8205](https://developer.mozilla.org/en/docs/Web/API/WindowSessionStorage)
JavaScript [WindowLocalStorage](https://developer.mozilla.org/en/docs/Web/API/WindowLocalStorage) to Kotlin

```kotlin
open external interface WindowLocalStorage {
    val localStorage: Storage
}

/**
 * Exposes the JavaScript [StorageEvent](https://developer.mozilla.org/en/docs/Web/API/StorageEvent) to Kotlin
 */
public external open class StorageEvent(type: String, eventInitDict: StorageEventInit = definedExternally) : Event {
    open val key: String?
    open val oldValue: String?
    open val newValue: String?
    open val url: String?
    open val storageArea: Storage?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE:
        val AT_TARGET:
        val BUBBLING_PHASE: Short
        val \n\n
    }
}

public external interface StorageEventInit : EventInit {
    var key: String? /* = null */
        get() = definedExternally
        set(value) =
    var oldValue: String? /* = null */
        get() = definedExternally
        set(value) =
    var newValue: String? /* = null */
        get() = definedExternally
        set(value) =
    var url: String? /* = "" */
        get() = definedExternally
        set(value) =
    var storageArea: Storage? /* = null */
        get() = definedExternally
        set(value) =
}

@kotlin.internal.InlineOnly
public inline fun StorageEventInit(key: String? = null,
    val o = js("({})")
    o["key"] = key
    o["oldValue"] = oldValue
    o["newValue"] = newValue
    o["url"] = url
    o["storageArea"] = storageArea
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
```

```kotlin
public external abstract class HTMLAppletElement : HTMLElement {
    open var align: String
    open var alt: String
    open var archive: String
    open var code:
    open var codeBase: String
    open var height: String
    open var hspace: Int
    open var name:
    open var _object: String
    open var vspace: Int
    open var width: String

    companion object {

        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE:
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short

        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

```kotlin
/**
 * Exposes the JavaScript [HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin
 */
public external abstract class HTMLMarqueeElement : HTMLElement {
    open var behavior: String
    open var bgColor: String
    open var direction: String
    open var height: String
    open var hspace: Int
    open var loop: Int
    open var scrollAmount: Int
    open var scrollDelay: Int
    open var trueSpeed: Boolean
    open var vspace: Int
    open var width: String
    fun start()
    fun stop()

    companion object {

        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE:
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```
```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [CustomEvent](https://developer.mozilla.org/en/docs/Web/API/CustomEvent) to Kotlin
 */
public external open class CustomEvent(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface CustomEventInit : EventInit {
        var detail: Any? /* = null */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        return o
    }

    public external interface EventListenerOptions {
        var capture: Boolean? /* = false */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun EventListenerOptions(capture: Boolean? = false): EventListenerOptions {
        val o = js("({})")
        o["capture"] = capture
        return o
    }

    public external interface AddEventListenerOptions : EventListenerOptions {
        var passive: Boolean? /* = false */
            get() = definedExternally
            set(value) = definedExternally
        var once: Boolean? /* = false */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["passive"] = passive
        o["once"] = once
        return o
    }

    public external interface NonElementParentNode {
        fun getElementById(elementId: String): Element?
    }

    /**
     * Exposes the JavaScript [DocumentOrShadowRoot](https://developer.mozilla.org/en/docs/Web/API/DocumentOrShadowRoot) to Kotlin
     */
    public external interface DocumentOrShadowRoot {
        valfullscreenElement: Element?
            get() = definedExternally
    }

    /**
     * Exposes the JavaScript [ParentNode](https://developer.mozilla.org/en/docs/Web/API/ParentNode) to Kotlin
     */
    public external interface ParentNode {
        val children: HTMLCollection
        valfirstElementChild: Element?
            get() = definedExternally
        vallastElementChild: Element?
            get() = definedExternally
        valchildElementCount: Int
        fun prepend(vararg nodes: dynamic): dynamic
        fun append(vararg nodes: dynamic): dynamic
        fun querySelector(selectors: String): Element?
        fun querySelectorAll(selectors: String): NodeList
    }

    /**
     * Exposes the JavaScript [NonDocumentTypeChildNode](https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode) to Kotlin
     */
    public external interface NonDocumentTypeChildNode {
        val previousElementSibling: Element?
            get() = definedExternally
        valnextElementSibling: Element?
            get() = definedExternally
    }

    /**
     * Exposes the JavaScript [ChildNode](https://developer.mozilla.org/en/docs/Web/API/ChildNode) to Kotlin
     */
    public external interface ChildNode {
        fun before(vararg nodes: dynamic): dynamic
        fun after(vararg nodes: dynamic): dynamic
        fun replaceWith(vararg nodes: dynamic): dynamic
        fun remove(): dynamic
    }

    /**
     * Exposes the JavaScript [Slotable](https://developer.mozilla.org/en/docs/Web/API/Slotable) to Kotlin
     */
    public external interface Slotable {
        valassignedSlot: HTMLSlotElement?
            get() = definedExternally
    }

    /**
     * Exposes the JavaScript [NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
     */
    public abstract class NodeList : ItemArrayLike<Node> {
        override fun item(index: Int): Node?
            asDynamic()[index]
    }

    @kotlin.internal.InlineOnly

    public external interface HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {
        override fun item(index: Int): Element?
            asDynamic()[index]
    }
```

Open Source Used In DNAC Platform Arica 8208
String): Element? = asDynamic()[name]

/**
 * Exposes the JavaScript [MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin
 */
public abstract open class MutationObserver(callback: (Array<MutationRecord>)? -> Unit) {
    fun observe(target: Node, options: MutationObserverInit = definedExternally) { fun disconnect() {
    fun takeRecords(): Array<MutationRecord>?
}

/**
 * Exposes the JavaScript [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var childList: NodeList
        get() = definedExternally
        set(value) = definedExternally
    var attributes: NodeList
        get() = definedExternally
        set(value) = definedExternally
    var characterData: NodeList
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    return o
}

/**
 * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin
 */
public external abstract class Node : EventTarget {
    val nodeType: Short
    val nodeName: String
    val baseURI: String
    val isConnected: Boolean
    val ownerDocument: Document?
    val parentNode: Node?
    val parentElement: Element?
    val childNodes: NodeList
    val firstChild: Node?
    val lastChild: Node?
    val previousSibling: Node?
    val nextSibling: Node?
    val nodeValue: String?
    val textContent: String?
    fun getRootNode(options: GetRootNodeOptions = definedExternally): Node
    fun hasChildNodes(): Boolean
    fun normalize()
    fun cloneNode(deep: Boolean = definedExternally): Node
    fun isEqualNode(otherNode: Node?): Boolean
    fun isSameNode(otherNode: Node?): Boolean
    fun compareDocumentPosition(other: Node): Short
    fun contains(other: Node?): Boolean
    fun lookupPrefix(namespace: String?): String?
    fun lookupNamespaceURI(prefix: String?): String?
    fun isDefaultNamespace(namespace: String?): Boolean
    fun insertBefore(node: Node, child: Node?): Node
    fun appendChild(node: Node): Node
    fun replaceChild(node: Node, child: Node): Node
    fun removeChild(child: Node): Node

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface GetRootNodeOptions {
    var composed: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean? = false): GetRootNodeOptions {

```kotlin
open val compatMode: String
open val characterSet: String
open val inputEncoding: String
open val contentType: String
open val doctype: DocumentType
open val documentElement: Element
open val location: Location
open val domain: String
open val referrer: String
var cookie: String
open val lastModified: String
open val readyState: DocumentReadyState
var title: String
var dir: String
var body: HTMLElement
open val head: HTMLHeadElement
open val images: HTMLCollection
open val embeds: HTMLCollection
open val plugins: HTMLCollection
open val links: HTMLCollection
open val forms: HTMLCollection
open val scripts: HTMLCollection
open val currentScript: HTMLOrSVGScriptElement
open val defaultView: Window
open val activeElement: Element
var designMode: String
var onreadystatechange: ((Event) -> dynamic)
```

**Open Source Used In DNAC Platform Arica**

8210

---

*Open Source Used In DNAC Platform Arica 8210*
onvolumechange: ((Event) -> dynamic)?
override var onwaiting: ((Event) -> dynamic)?
override var ongpointercapture: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointermove: ((PointerEvent) -> dynamic)?
override var onpointerdown: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy: ((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override var fullscreenElement: Element?
override val childElementCount: Int
fun getElementsByTagName(qualifiedName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun getElementsByClassName(classNames: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun createElementNS(namespace: String?, qualifiedName: String, options: ElementCreationOptions = definedExternally): Element
fun createDocumentFragment(): DocumentFragment
fun createTextNode(data: String): Text
fun createCDATASection(data: String): CDATASection
fun createComment(data: String): Comment
fun createProcessingInstruction(target: String, data: String): ProcessingInstruction
fun importNode(node: Node, deep: Boolean = definedExternally): Node
fun adoptNode(node: Node): Node
fun createAttribute(localName: String): Attr
fun createAttributeNS(namespace: String?, qualifiedName: String): Attr
fun createEvent(`interface`: String): Event
fun createRange(): Range
fun createNodeIterator(root: Node, whatToShow: Int = definedExternally, filter: ((Node) -> Short)? = definedExternally): NodeIterator
fun createTreeWalker(root: Node, whatToShow: Int = definedExternally, filter: NodeFilter? = definedExternally): TreeWalker
fun createTreeWalker(root: Node, whatToShow: Int = definedExternally, filter: ((Node) -> Short)? = definedExternally): TreeWalker
fun getElementsByName(elementName: String): NodeList
fun open(type: String = definedExternally, replace: String = definedExternally): Document
fun open(url: String, name: String, features: String): Window
fun close()
fun write(vararg text: String)
fun writeln(vararg text: String)
fun hasFocus(): Boolean
fun execCommand(commandId: String, showUI: Boolean = definedExternally, value: String = definedExternally): Boolean
fun queryCommandEnabled(commandId: String): Boolean
fun queryCommandIndeterm(commandId: String): Boolean
fun queryCommandState(commandId: String): Boolean
fun queryCommandSupported(commandId: String): Boolean
fun queryCommandValue(commandId: String): Boolean
fun createElementById(elementId: String): Element?
fun prepend(vararg nodes: dynamic)
fun append(vararg nodes: dynamic)
fun querySelector(selectors: String): Element?
fun querySelectorAll(selectors: String): NodeList
fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMPoint
fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
fun override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMRect
fun override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
fun companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE:
Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public inline operator fun Document.get(name: String): dynamic = asDynamic()[name]

/**
 * Exposes the JavaScript [XMLDocument](https://developer.mozilla.org/en/docs/Web/API/XMLDocument) to Kotlin
 */
public external open class XMLDocument : Document {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface ElementCreationOptions {
    var `is`: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementCreationOptions(`is`: String? = undefined): ElementCreationOptions {
    val o = js("({})")
    o["is"] = `is`
    return o
}

/**
 * Exposes the JavaScript [DOMImplementation](https://developer.mozilla.org/en/docs/Web/API/DOMImplementation) to Kotlin
 */
public external abstract class DOMImplementation {
    fun createDocumentType(qualifiedName: String, publicId: String, systemId: String): DocumentType
    fun createDocument(namespace: String?, qualifiedName: String, doctype: DocumentType? = definedExternally): XMLDocument
    fun createHTMLDocument(title: String = definedExternally): Document
    fun hasFeature(): Boolean
}

/**
 * Exposes the JavaScript [DocumentType](https://developer.mozilla.org/en/docs/Web/API/DocumentType) to Kotlin
 */
public external abstract class DocumentType : Node, ChildNode {
    open val name: String
    open val publicId: String
    open val systemId: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [DocumentFragment](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) to Kotlin
 */
public external open class DocumentFragment : Node, NonElementParentNode, ParentNode {
    override val children: HTMLCollection
    override val firstElementChild: Element?
    override val lastElementChild: Element?
    override val childElementCount: Int
    override fun getElementById(elementId: String): Element?
    override fun prepend(vararg nodes: dynamic)
    override fun append(vararg nodes: dynamic)
    override fun querySelector(selectors: String): Element?
    override fun querySelectorAll(selectors: String): NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin
 */
public external open
class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {
    open val mode: ShadowRootMode
    open val host: Element
    override val fullscreenElement: Element?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin
 */
public external abstract class
Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils,
UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent,
UnionElementOrProcessingInstruction {
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val tagName: String
    open var id: String
    open var className: String
    open val classList: DOMTokenList
    open var slot: String
    open val attributes: NamedNodeMap
    open val shadowRoot: ShadowRoot?
    open var scrollTop: Double
    open var scrollLeft: Double
    open val scrollWidth: Int
    open val scrollHeight: Int
    open val clientTop: Int
    open val clientLeft: Int
    open val clientWidth: Int
    open val clientHeight: Int
    open val innerHTML: String
    open val outerHTML: String
    fun hasAttributes(): Boolean
    fun getAttributeNames(): Array<String>
    fun getAttribute(qualifiedName: String): String?
    fun getAttributeNS(namespace: String?, localName: String): String?
    fun setAttribute(qualifiedName: String, value: String)
    fun setAttributeNS(namespace: String?, qualifiedName: String, value: String)
    fun removeAttribute(qualifiedName: String)
    fun removeAttributeNS(namespace: String?, localName: String)
    fun hasAttribute(qualifiedName: String): Boolean
    fun hasAttributeNS(namespace: String?, localName: String): Boolean
    fun getAttributeNode(qualifiedName: String): Attr?
    fun getAttributeNodeNS(namespace: String?, localName: String): Attr?
    fun setAttributeNode(attr: Attr): Attr
    fun setAttributeNodeNS(attr: Attr): Attr
    fun attachShadow(init: ShadowRootInit): ShadowRoot
    fun closest(selectors: String): Element?
    fun matches(selectors: String): Boolean
    fun webkitMatchesSelector(selectors: String): Boolean
    fun getElementsByTagName(qualifiedName: String): HTMLCollection
    fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
    fun getElementsByClassName(classNames: String): HTMLCollection
    fun getElementsByClassNameNS(namespace: String?, localName: String): HTMLCollection
    fun getElementsByTagClassName(classNames: String): HTMLCollection
    fun getElementsByTagNameNS(where: String, element: Element?): HTMLCollection
    fun insertAdjacentElement(where: String, data: String): String
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun scrollIntoView(): Boolean
    fun scrollIntoView(arg: dynamic): Boolean
    fun scroll(overflow: ScrollToOptions = definedExternally): Boolean
    fun scroll(x: Double, y: Double): Boolean
    fun scrollBy(options: ScrollToOptions = definedExternally): Boolean
    fun scrollBy(x: Double, y: Double): Boolean
    fun insertAdjacentHTML(position: String, text: String): Boolean
    fun pointerCapture(pointerId: Int): Boolean
    fun releasePointerCapture(pointerId: Int): Boolean
    fun requestFullScreen():
Promise<Unit>

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

    public external interface ShadowRootInit {
        var mode: ShadowRootMode?
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun ShadowRootInit(mode: ShadowRootMode?): ShadowRootInit {
        val o = js("({})")
        o["mode"] = mode
        return o
    }

    /**
     * Exposes the JavaScript [NamedNodeMap](https://developer.mozilla.org/en/docs/Web/API/NamedNodeMap) to
     * Kotlin
     */
    public external abstract class NamedNodeMap : ItemArrayLike<Attr> {
        fun getNamedItemNS(namespace: String?, localName: String): Attr?
        fun setNamedItem(attr: Attr): Attr?
        fun setNamedItemNS(attr: Attr): Attr?
        fun removeNamedItem(qualifiedName: String): Attr
        fun removeNamedItemNS(namespace: String?, localName: String): Attr
        fun getNamedItem(qualifiedName: String): Attr
        fun item(index: Int): Attr?
    }

    @kotlin.internal.InlineOnly
    public inline operator fun NamedNodeMap.get(index: Int): Attr? = asDynamic()[index]

    @kotlin.internal.InlineOnly
    public inline operator fun NamedNodeMap.get(qualifiedName: String): Attr? = asDynamic()[qualifiedName]

    /**
     * Exposes the JavaScript [Attr](https://developer.mozilla.org/en/docs/Web/API/Attr) to Kotlin
     */
    public external abstract class Attr : Node {
        open val namespaceURI: String?
        open val prefix: String?
        open val localName: String
        open val name: String
        open var value: String
        open val ownerElement: Element?
        open val specified: Boolean

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }

        /**
         * Exposes the JavaScript [CharacterData](https://developer.mozilla.org/en/docs/Web/API/CharacterData) to Kotlin
         */
        public external abstract class CharacterData : Node, NonDocumentTypeChildNode, ChildNode {
            open var data: String
            open val length: Int
            fun substringData(offset: Int, count: Int): String
            fun appendData(data: String)
            fun insertData(offset: Int, data: String)
            fun deleteData(offset: Int, count: Int)
            fun replaceData(offset: Int, count: Int, data: String)

            companion object {
                val ELEMENT_NODE: Short
                val ATTRIBUTE_NODE: Short
                val TEXT_NODE: Short
                val CDATA_SECTION_NODE: Short
                val ENTITY_REFERENCE_NODE: Short
                val ENTITY_NODE: Short
                val PROCESSING_INSTRUCTION_NODE: Short
                val COMMENT_NODE: Short
                val DOCUMENT_NODE: Short
                val DOCUMENT_TYPE_NODE: Short
                val DOCUMENT_FRAGMENT_NODE: Short
                val NOTATION_NODE: Short
                val DOCUMENT_POSITION_DISCONNECTED: Short
                val DOCUMENT_POSITION_PRECEDING: Short
                val DOCUMENT_POSITION_FOLLOWING: Short
                val DOCUMENT_POSITION_CONTAINS: Short
                val DOCUMENT_POSITION_CONTAINED_BY: Short
                val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
            }

            /**
             * Exposes the JavaScript [Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin
             */
            public external open class Text(data: String, nodeType: Int = Text.TEXT_NODE)
            fun substringData(offset: Int, count: Int): String
            fun appendData(data: String)
            fun insertData(offset: Int, data: String)
            fun deleteData(offset: Int, count: Int)
            fun replaceData(offset: Int, count: Int, data: String)
            fun setData(data: String)
        }
    }
}
Open Source Used In DNAC Platform Arica 8215

String = definedExternally) : CharacterData, Slotable, GeometryUtils {
    open val wholeText: String
    override val assignedSlot: HTMLSlotElement?
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    fun splitText(offset: Int): Text?
    override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
    override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin
 */
public external
open class CDATASection : Text {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin
 */
public external abstract class ProcessingInstruction : CharacterData, LinkStyle,
UnionElementOrProcessingInstruction {
    open val target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin
 */
public external open class Comment(data: String = definedExternally) : CharacterData {
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
    }
}
Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
 */
public external open class Range {
    open val startContainer: Node
    open val startOffset: Int
    open val endContainer: Node
    open val endOffset: Int
    fun setStart(node: Node, offset: Int)
    fun setEnd(node: Node, offset: Int)
    fun setStartBefore(node: Node)
    fun setStartAfter(node: Node)
    fun setEndBefore(node: Node)
    fun setEndAfter(node: Node)
    fun collapse(toStart: Boolean = definedExternally)
    fun selectNode(node: Node)
    fun selectNodeContents(node: Node)
    fun compareBoundaryPoints(how: Range, otherRange: Range): Short
    fun deleteContents()
    fun extractContents(): DocumentFragment
    fun cloneContents(): DocumentFragment
    fun insertNode(node: Node)
    fun surroundContents(newParent: Node)
    fun cloneRange(): Range
    fun detach()
    fun isPointInRange(node: Node, offset: Int): Boolean
    fun comparePoint(node: Node, offset: Int): Short
    fun intersectsNode(node: Node): Boolean
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun createContextualFragment(fragment: String): DocumentFragment
}

companion object {
    val START_TO_START: Short
    val START_TO_END: Short
    val END_TO_END: Short
    val END_TO_START: Short
}

/**
 * Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
    open val root: Node
    open val referenceNode: Node
    open val pointerBeforeReferenceNode: Boolean
    open val whatToShow: Int
    open val filter: NodeFilter?
    fun nextNode(): Node?
    fun previousNode(): Node?
    fun detach()
}

/**
 * Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
    open val root: Node
    open val whatToShow: Int
    open val filter: NodeFilter?
    open var currentNode: Node
    fun parentNode(): Node?
    fun firstChild(): Node?
    fun lastChild(): Node?
    fun previousSibling(): Node?
    fun nextSibling(): Node?
    fun previousNode(): Node?
    fun nextNode(): Node?
}

/**
 * Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
    fun acceptNode(node: Node): Short

    companion object {
        val FILTER_ACCEPT: Short
        val FILTER_REJECT: Short
        val FILTER_SKIP: Short
        val SHOW_ALL: Int
        val SHOW_ELEMENT: Int
        val SHOW_ATTRIBUTE: Int
        val SHOW_TEXT: Int
        val SHOW_CDATA_SECTION: Int
        val SHOW_ENTITY_REFERENCE: Int
        val SHOW_ENTITY: Int
        val SHOW_PROCESSING_INSTRUCTION: Int
    }
}

/**
 * Exposes the JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
 */
public external abstract class DOMTokenList : ItemArrayLike<String> {
    open var value: String
    fun contains(token: String): Boolean
    fun add(vararg tokens: String)
    fun remove(vararg tokens: String)
    fun toggle(token: String, force: Boolean = definedExternally): Boolean
    fun replace(token: String, newToken: String)
    fun supports(token: String): Boolean
    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String = asDynamic()[index]
open val y: Double
    open val z: Double
    open val w: Double
    fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint
}

/**
 * Exposes the JavaScript [DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin
 */
public external open class DOMPoint : DOMPointReadOnly {
    constructor(point: DOMPointInit)
    constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)

    override var x: Double
    override var y: Double
    override var z: Double
    override var w: Double
}

/**
 * Exposes the JavaScript [DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin
 */
public external interface DOMPointInit {
    var x: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var y: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var z: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var w: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["w"] = w
    return o
}

/**
 * Exposes the JavaScript [DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin
 */
public external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {
    override var x: Double
    override var y: Double
    override var width: Double
    override var height: Double
}

/**
 * Exposes the JavaScript [DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin
 */
public external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {
    open val x: Double
    open val y: Double
    open val width: Double
    open val height: Double
    open val top: Double
    open val right: Double
    open val bottom: Double
    open val left: Double
}

public external interface DOMRectList : ItemArrayLike<DOMRect> {
    override fun item(index: Int): DOMRect?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]

/**
 * Exposes the JavaScript [DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin
 */
public external open class DOMQuad {
    constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)
    constructor(rect: DOMRectInit)

    open val p1: DOMPoint
    open val p2: DOMPoint
    open val p3: DOMPoint
    open val p4: DOMPoint

    open val bounds: DOMRectReadOnly
}

/**
 * Exposes the JavaScript [DOMMatrixReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMMatrixReadOnly) to Kotlin
 */
public external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {
    open val a: Double
    open val b: Double
    open val c: Double
    open val d: Double
    open val e: Double
    open val f: Double

    open val m11: Double
    open val m12: Double
    open val m13: Double
    open val m14: Double
    open val m21: Double
    open val m22: Double
    open val m23: Double
    open val m24: Double
    open val m31: Double
    open val m32: Double
    open val m33: Double
    open val m34: Double
    open val m41: Double
    open val m42: Double
    open val m43: Double
    open val m44: Double
    open val is2D: Boolean
    open val isIdentity: Boolean
    fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
}
Double = definedExternally): DOMMatrix\n    fun scaleNonUniform(scaleX: Double, scaleY: Double = scaleZ = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix\n    fun rotate(angle: Double, originX: Double, originY: Double = definedExternally): DOMMatrix\n    fun rotateFromVector(x: Double, y: Double): DOMMatrix\n    fun rotateAxisAngle(x: Double, y: Double, z: Double, angle: Double): DOMMatrix\n    fun skewX(sx: Double): DOMMatrix\n    fun skewY(sy: Double): DOMMatrix
    fun multiply(other: DOMMatrix): DOMMatrix\n    fun flipX(): DOMMatrix\n    fun flipY(): DOMMatrix\n    fun inverse(): DOMMatrix\n    fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint\n    fun toFloat32Array(): Float32Array\n    fun toFloat64Array(): Float64Array

/**
 * Exposes the JavaScript [DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin
 */
public external open class DOMMatrix() : DOMMatrixReadOnly {
    constructor(transformList: String)
    constructor(other: DOMMatrixReadOnly)
    constructor(array32: Float32Array)
    constructor(array64: Float64Array)
    constructor(numberSequence: Array<Double>)
    override var a: Double
    override var b: Double
    override var c: Double
    override var d: Double
    override var e: Double
    override var f: Double
    override var m11: Double
    override var m12: Double
    override var m13: Double
    override var m14: Double
    override var m21: Double
    override var m22: Double
    override var m23: Double
    override var m24: Double
    override var m31: Double
    override var m32: Double
    override var m33: Double
    override var m34: Double
    override var m41: Double
    override var m42: Double
    override var m43: Double
    override var m44: Double
    fun multiplySelf(other: DOMMatrix): DOMMatrix
    fun preMultiplySelf(other: DOMMatrix): DOMMatrix
    fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3dSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVectorSelf(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngleSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
    fun skewXSelf(sx: Double): DOMMatrix
    fun skewYSelf(sy: Double): DOMMatrix
    fun invertSelf(): DOMMatrix
    fun setMatrixValue(transformList: String): DOMMatrix
}

/**
 * Exposes the JavaScript [ScrollOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollOptions) to Kotlin
 */
public external interface ScrollOptions {
    var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {
    val o = js("({})")
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
    open val media: String
    open val matches: Boolean
    open var onchange: ((Event) -> dynamic)?
    fun addListener(listener: EventListener?)
    fun addListener(listener: ((Event) -> Unit)?)
    fun removeListener(listener: EventListener?)
    fun removeListener(listener: ((Event) -> Unit)?)
}

/**
 * Exposes the JavaScript [MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin
 */
public external class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event

public external interface MediaQueryListEventInit : EventInit {
    var

    public external interface BoxQuadOptions {
        var box: CSSBoxType? /* = CSSBoxType.BORDER */
            get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun BoxQuadOptions(box: CSSBoxType? = CSSBoxType.BORDER, relativeTo: dynamic = undefined): BoxQuadOptions {
        val o = js("({})")
        o["box"] = box
        o["relativeTo"] = relativeTo
        return o
    }

    public external interface ConvertCoordinateOptions {
        var fromBox: CSSBoxType? /* = CSSBoxType.BORDER */
            get() = definedExternally
        set(value) = definedExternally
        var toBox: CSSBoxType? /* = CSSBoxType.BORDER */
            get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun ConvertCoordinateOptions(fromBox: CSSBoxType? = CSSBoxType.BORDER, toBox: CSSBoxType? = CSSBoxType.BORDER): ConvertCoordinateOptions {
        val o = js("({})")
        o["fromBox"] = fromBox
        o["toBox"] = toBox
        return o
    }

    @kotlin.internal.InlineOnly
    public inline fun BoxQuadOptions.get(): BoxQuadOptions {
        return this
    }

    @kotlin.internal.InlineOnly
    public inline fun ConvertCoordinateOptions.get(): ConvertCoordinateOptions {
        return this
    }

    /**
     * Exposes the JavaScript GeometryUtils to Kotlin
     */
    public external interface GeometryUtils {
        fun getBoxQuads(options: BoxQuadOptions = definedExternally): Array<DOMQuad>
        fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
        fun convertRectFromNode(rect: DOMRect, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMRect
        fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMPoint
    }

    /**
     * Exposes the JavaScript Touch to Kotlin
     */
    public external open class TouchEvent : UIEvent {
        open val touches: TouchList
        open val targetTouches: TouchList
        open val changedTouches: TouchList
        open val altKey: Boolean
        open val metaKey: Boolean
        open val ctrlKey: Boolean
        open val shiftKey: Boolean
        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
        }
    }

    @kotlin.internal.InlineOnly
    public inline operator fun TouchList.get(index: Int): Touch? = asDynamic()[index]

    Open Source Used In DNAC Platform Arica 8219

fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external open class Audio(src: String = definedExternally) : HTMLAudioElement {
    override var onabort: ((Event) -> dynamic)?
    override var onblur: ((FocusEvent) -> dynamic)?
    override var oncancel: ((Event) -> dynamic)?
    override var oncanplay: ((Event) -> dynamic)?
    override var oncanplaythrough: ((Event) -> dynamic)?
    override var onchange: ((Event) -> dynamic)?
    override var onclick: ((MouseEvent) -> dynamic)?
    override var onclose: ((Event) -> dynamic)?
    override var oncontextmenu: ((MouseEv
HTMLOrSVGImageElement : CanvasImageSource

public external interface CanvasImageSource

ImageBitmapSource

public external interface ImageBitmapSource

UnionMessagePortOrWindowProxy

public external interface HTMLOrSVGImageElement

public external interface CanvasImageSource : ImageBitmapSource

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface DocumentReadyState {
    companion object
}

"loading".asDynamic().unsafeCast<DocumentReadyState>()

"interactive".asDynamic().unsafeCast<DocumentReadyState>()

"complete".asDynamic().unsafeCast<DocumentReadyState>()

CanPlayTypeResult {
    companion object
}

canPlayTypeResult.Companion.EMPTY: CanPlayTypeResult get() = "".asDynamic().unsafeCast<CanPlayTypeResult>()

canPlayTypeResult.Companion.MAYBE: CanPlayTypeResult get() = 
"maybe".asDynamic().unsafeCast<CanPlayTypeResult>()

canPlayTypeResult.Companion.PROBABLY: CanPlayTypeResult get() = 
"probably".asDynamic().unsafeCast<CanPlayTypeResult>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface TextTrackMode {
    companion object
}

textTrackMode.Companion.DISABLED: TextTrackMode get() = 
"disabled".asDynamic().unsafeCast<TextTrackMode>()

textTrackMode.Companion.HIDDEN: TextTrackMode get() = 
"hidden".asDynamic().unsafeCast<TextTrackMode>()

textTrackMode.Companion.SHOWING: TextTrackMode get() = 
"showing".asDynamic().unsafeCast<TextTrackMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface TextTrackKind {
    companion object
}

textTrackKind.Companion.SUBTITLES: TextTrackKind get() = 
"subtitles".asDynamic().unsafeCast<TextTrackKind>()

textTrackKind.Companion.CAPTIONS: TextTrackKind get() = 
"captions".asDynamic().unsafeCast<TextTrackKind>()

textTrackKind.Companion.DESCRIPTIONS: TextTrackKind get() = 
"descriptions".asDynamic().unsafeCast<TextTrackKind>()

textTrackKind.Companion.CHAPTERS: TextTrackKind get() = 
"chapter".asDynamic().unsafeCast<TextTrackKind>()

textTrackKind.Companion.METADATA: TextTrackKind get() = 
"metadata".asDynamic().unsafeCast<TextTrackKind>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SelectionMode {
    companion object
}

selectionMode.Companion.SELECT: SelectionMode get() = 
"select".asDynamic().unsafeCast<SelectionMode>()

selectionMode.Companion.START: SelectionMode get() = 
"start".asDynamic().unsafeCast<SelectionMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasFillRule {
    companion object
}

canvasFillRule.Companion.NONZERO: CanvasFillRule get() = 
"nonzero".asDynamic().unsafeCast<CanvasFillRule>()

canvasFillRule.Companion.EVENODD: CanvasFillRule get() = 
"evenodd".asDynamic().unsafeCast<CanvasFillRule>()

/* please, don't implement this interface! */
public external interface ImageSmoothingQuality {
    companion object
}


public external interface CanvasLineCap {
    companion object
}

public inline val CanvasLineCap.Companion.BUTT: CanvasLineCap get() = "butt".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.ROUND: CanvasLineCap get() = "round".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.SQUARE: CanvasLineCap get() = "square".asDynamic().unsafeCast<CanvasLineCap>()

public external interface CanvasLineJoin {
    companion object
}

public inline val CanvasLineJoin.Companion.ROUND: CanvasLineJoin get() = "round".asDynamic().unsafeCast<CanvasLineJoin>()

public inline val CanvasLineJoin.Companion.BEVEL: CanvasLineJoin get() = "bevel".asDynamic().unsafeCast<CanvasLineJoin>()

public inline val CanvasLineJoin.Companion.MITER: CanvasLineJoin get() = "miter".asDynamic().unsafeCast<CanvasLineJoin>()

public external interface CanvasTextAlign {
    companion object
}

public inline val CanvasTextAlign.Companion.START: CanvasTextAlign get() = "start".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.END: CanvasTextAlign get() = "end".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.LEFT: CanvasTextAlign get() = "left".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.RIGHT: CanvasTextAlign get() = "right".asDynamic().unsafeCast<CanvasTextAlign>()

public inline val CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() = "center".asDynamic().unsafeCast<CanvasTextAlign>()

public external interface CanvasDirection {
    companion object
}

public inline val CanvasDirection.Companion.LTR: CanvasDirection get() = "ltr".asDynamic().unsafeCast<CanvasDirection>()

public inline val CanvasDirection.Companion.RTL: CanvasDirection get() = "rtl".asDynamic().unsafeCast<CanvasDirection>()

public inline val CanvasDirection.Companion.INHERIT: CanvasDirection get() = "inherit".asDynamic().unsafeCast<CanvasDirection>()

public external interface ScrollRestoration
public inline val ScrollRestoration.Companion.AUTO: ScrollRestoration get() = "auto".asDynamic().unsafeCast<ScrollRestoration>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ImageOrientation {
    companion object
}


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface PremultiplyAlpha {
    companion object
}

public inline val PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() = "none".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() = "premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() = "default".asDynamic().unsafeCast<PremultiplyAlpha>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ColorSpaceConversion {
    companion object
}

public inline val ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() = "none".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() = "default".asDynamic().unsafeCast<ColorSpaceConversion>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResizeQuality {
    companion object
}

public inline val ResizeQuality.Companion.PIXELATED: ResizeQuality get() = "pixelated".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.LOW: ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.MEDIUM: ResizeQuality get() = "medium".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.HIGH: ResizeQuality get() = "high".asDynamic().unsafeCast<ResizeQuality>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface BinaryType {
    companion object
}

public inline val BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()

public inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() = "arraybuffer".asDynamic().unsafeCast<BinaryType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WorkerType {
    companion object
}

public inline val WorkerType.Companion.CLASSIC: WorkerType get() = "classic".asDynamic().unsafeCast<WorkerType>()

public inline val WorkerType.Companion.MODULE: WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ShadowRootMode {
    companion object
}

public inline val ShadowRootMode.Companion.OPEN: ShadowRootMode get() = "open".asDynamic().unsafeCast<ShadowRootMode>()

public inline val ShadowRootMode.Companion.CLOSED: ShadowRootMode get() = "closed".asDynamic().unsafeCast<ShadowRootMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollBehavior {
    companion object
}

public inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() = "auto".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() = "instant".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() = "smooth".asDynamic().unsafeCast<ScrollBehavior>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollLogicalPosition {
    companion object
}

public inline val ScrollLogicalPosition.Companion.START: ScrollLogicalPosition.get() =
public inline val ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() = "center".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() = "end".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.NEAREST: ScrollLogicalPosition get() = "nearest".asDynamic().unsafeCast<ScrollLogicalPosition>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CSSBoxType {

    companion object
}

public inline val CSSBoxType.Companion.MARGIN: CSSBoxType get() = "margin".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.BORDER: CSSBoxType get() = "border".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.PADDING: CSSBoxType get() = "padding".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.CONTENT: CSSBoxType get() = "content".asDynamic().unsafeCast<CSSBoxType>()
var keepalive: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var window:
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RequestInit(method: String? = undefined,
headers: dynamic = undefined, body: dynamic = undefined, referrer: String? = undefined, referrerPolicy: dynamic = undefined,
    val o = js("({})")
    o["method"] = method
    o["headers"] = headers
    o["body"] = body
    o["referrer"] = referrer
    o["referrerPolicy"] = referrerPolicy
    o["mode"] = mode
    o["credentials"] = credentials
    o["cache"] = cache
    o["redirect"] = redirect
    o["integrity"] = integrity
    o["keepalive"] = keepalive
    o["window"] = window
    override fun clone(): Response
        get() = this
    override fun arrayBuffer(): Promise<ArrayBuffer>
        get() = this
    override fun blob(): Promise<Blob>
        get() = this
    override fun formData(): PromiseFormData
        get() = this
    override fun json(): Promise<Any?>
        get() = this
    override fun text(): Promise<String>
        get() = this
}

/**
 * Exposes the JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to
 * Kotlin
 */
public external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body {
    open val type: ResponseType
    open val url: String
    open val redirected:
    open val status: Short
    open val ok: Boolean
    open val statusText: String
    open val headers:
    open val body: dynamic
    open val trailer: Promise<Headers>
    override val bodyUsed: Boolean
    fun clone(): Response
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun formData(): PromiseFormData
    override fun json(): Promise<Any?>
    override fun text(): Promise<String>
}

public external interface ResponseInit {
    var status: Short? /* = 200 */
        get() = definedExternally
        set(value) = definedExternally
    var statusText:
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ResponseInit(status: Short? = 200,
statusText: String? = "OK", headers: dynamic = undefined): ResponseInit {
    val o = js("({})")
    o["status"] = status
    o["statusText"] = statusText
    o["headers"] = headers
    override fun override fun blob(): Promise<Blob>
    

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface

/* please, don't implement
 * this interface!
 */
@Suppress("NESTED_CLASS_INternal. In DNAC Platform Arica 8228
"image".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.MANIFEST: RequestDestination get() = "manifest".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.MEDIA: RequestDestination get() = "media".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.OBJECT: RequestDestination get() = "object".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.REPORT: RequestDestination get() = "report".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SCRIPT: RequestDestination get() = "script".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SERVICEWORKER: RequestDestination get() = "serviceworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SHAREDWORKER: RequestDestination get() = "sharedworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.STYLE: RequestDestination get() = "style".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.WORKER: RequestDestination get() = "worker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.XSLT: RequestDestination get() = "xslt".asDynamic().unsafeCast<RequestDestination>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestMode {

    companion object
}

public inline val RequestMode.Companion.NAVIGATE: RequestMode get() = "navigate".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.SAME_ORIGIN: RequestMode get() = "same-origin".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.NO_CORS: RequestMode get() = "no-cors".asDynamic().unsafeCast<RequestMode>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCredentials {

    companion object
}

public inline val RequestCredentials.Companion.OMIT: RequestCredentials get() = "omit".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.SAME_ORIGIN: RequestCredentials get() = "same-origin".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.INCLUDE: RequestCredentials get() = "include".asDynamic().unsafeCast<RequestCredentials>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCache {

    companion object
}

public inline val RequestCache.Companion.DEFAULT: RequestCache get() = "default".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_STORE: RequestCache get() = "no-store".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.RELOAD: RequestCache get() = "reload".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_CACHE: RequestCache get() = "no-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.FORCE_CACHE: RequestCache get() = "force-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = "only-if-cached".asDynamic().unsafeCast<RequestCache>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestRedirect
Open Source Used In DNAC Platform Arica 8230

/*
 * Copyright 2010-2020 JetBrains s.r.o. and
 * Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that
 * can be found in the license/LICENSE.txt file.
 * DO NOT EDIT! See github.com/kotlin/dukat for details
 * Exposes the JavaScript MediaStream() to Kotlin
 */

package org.w3c.dom.mediacapture

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript MediaStream to Kotlin
 */
public external open class MediaStream() : EventTarget, MediaProvider {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}

/**
 * Exposes the JavaScript MediaStreamTrack to Kotlin
 */
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open var muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop()
    fun getCapabilities(): MediaTrackCapabilities
    fun getSettings(): MediaTrackSettings
    fun applyConstraints(constraints: MediaTrackSupportedConstraints = definedExternally): Promise<Unit>
}

/**
 * Exposes the JavaScript MediaTrackSupportedConstraints to Kotlin
 */
public external interface MediaTrackSupportedConstraints {
    var width: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var height: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}
definedExternally
set(value) = definedExternally
var volume: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var sampleRate: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var sampleSize: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var echoCancellation: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var autoGainControl: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var noiseSuppression: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var latency: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var channelCount: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var deviceId: Boolean? /* = true */
get() =
definedExternally
set(value) = definedExternally
var groupId: Boolean? /* = true */
get() =

MediaTrackSupportedConstraints(width: Boolean? = true, height: Boolean? = true, aspectRatio: Boolean? = true,
frameRate: Boolean? = true, facingMode: Boolean? = true, resizeMode: Boolean? = true, volume: Boolean? = true,
sampleRate: Boolean? = true, sampleSize: Boolean? = true, echoCancellation: Boolean? = true, autoGainControl:
deviceId: Boolean? = true, groupId: Boolean? = true): MediaTrackSupportedConstraints {
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

@kotlin.internal.InlineOnly
public inline fun MediaTrackCapabilities(width: ULongRange? = undefined, height: ULongRange? = undefined,
aspectRatio: DoubleRange? = undefined, frameRate: DoubleRange? = undefined, facingMode: Array<String>? = undefined,
resizeMode: Array<String>? = undefined, volume: Boolean? = undefined, sampleRate: ULongRange? = undefined,
sampleSize: ULongRange? = undefined, echoCancellation: Array<Boolean>? = undefined, autoGainControl:
Array<Boolean>? = undefined, noiseSuppression: Array<Boolean>? = undefined, latency: DoubleRange? = undefined,
channelCount: ULongRange? = undefined, deviceId: String? = undefined, groupId: String? = undefined): MediaTrackCapabilities {
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    return o
}
open source used in DNAC platform Arica

```kotlin
/**
 * Exposes the JavaScript [MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin
 */
public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced: Array<MediaTrackConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var volume: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var latency: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var groupId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    return o
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["advanced"] = advanced
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}
```

```kotlin
/**
 * Exposes the JavaScript [MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var volume: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var latency: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var groupId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    return o
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}
```
aspectRatio: Double?
    get() = definedExternally
    set(value) = definedExternally

var frameRate:
    get() = definedExternally
    set(value) = definedExternally

var facingMode: String?
    get() = definedExternally
    set(value) = definedExternally

var resizeMode: String?
    get() = definedExternally
    set(value) = definedExternally

var volume: Double?
    get() = definedExternally
    set(value) = definedExternally

var sampleRate: Int?
    get() = definedExternally
    set(value) = definedExternally

var sampleSize: Int?
    get() = definedExternally
    set(value) = definedExternally

var echoCancellation: Boolean?
    get() = definedExternally
    set(value) = definedExternally

var autoGainControl: Boolean?
    get() = definedExternally
    set(value) = definedExternally

var noiseSuppression: Boolean?
    get() = definedExternally
    set(value) = definedExternally

var latency:
    get() = definedExternally
    set(value) = definedExternally

var channelCount: Int?
    get() = definedExternally
    set(value) = definedExternally

var deviceId: String?
    get() = definedExternally
    set(value) = definedExternally

var groupId: String?
    get() = definedExternally
    set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript
[MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin
*/
public external open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) : Event {
    open val track: MediaStreamTrack

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaStreamTrackEventInit : EventInit {
    var track: MediaStreamTrack?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["track"] = track
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin
*/
public external abstract class MediaDevices : EventTarget {
    open var ondevicechange: ((Event) -> dynamic)?
    fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>
    fun getSupportedConstraints(): MediaTrackSupportedConstraints
    fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>
}

/**
 * Exposes the JavaScript
[MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin
*/
public...
external abstract class MediaDeviceInfo {
    open val deviceId: String
    open val kind: MediaDeviceKind
    open val label: String
    open val groupId: String

    fun toJSON(): dynamic
}

public external abstract class InputDeviceInfo : MediaDeviceInfo {
    fun getCapabilities(): MediaTrackCapabilities
}

/**
 * Exposes the JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin
 */
public external interface MediaStreamConstraints {
    var video: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var audio: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
    val o = js("({})")
    o["video"] = video
    o["audio"] = audio
    return o
}

public external interface ConstrainablePattern {
    var onoverconstrained: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    fun getCapabilities(): Capabilities
    fun getConstraints(): Constraints
    fun getSettings(): Settings
    fun applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>
}

/**
 * Exposes the JavaScript [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin
 */
public external interface DoubleRange {
    var max: Double?
        get() = definedExternally
        set(value) = definedExternally
    var min: Double?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
    val o = js("({})")
    o["max"] = max
    o["min"] = min
    return o
}

public external interface ConstrainDoubleRange : DoubleRange {
    var exact: Double?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Double?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    o["max"] = max
    o["min"] = min
    return o
}

public external interface ULongRange {
    var max: Int?
        get() = definedExternally
        set(value) = definedExternally
    var min: Int?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ULongRange(max: Int? = undefined, min: Int? = undefined): ULongRange {
    val o = js("({})")
    o["max"] = max
    o["min"] = min
    return o
}

public external interface ConstrainULongRange : ULongRange {
    var exact: Int?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Int?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    o["max"] = max
    o["min"] = min
    return o
}

/**
 * Exposes the JavaScript [ConstrainBooleanParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainBooleanParameters) to Kotlin
 */
public external interface ConstrainBooleanParameters {
    var exact: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Boolean?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}

/**
 * Exposes the JavaScript [ConstrainDOMStringParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainDOMStringParameters) to Kotlin
 */
public external interface ConstrainDOMStringParameters {
    var exact: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var ideal: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConstrainDOMStringParameters(exact: dynamic = undefined, ideal: dynamic = undefined): ConstrainDOMStringParameters {
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}

public external interface Capabilities

@kotlin.internal.InlineOnly
public inline fun Capabilities(): Capabilities {
    val o = js("({})")
    return o
}
Settingsisateur.kotlin.internal.InlineOnly

public inline fun Settings(): Settings {
    val o = js("({})")
    return o
}

public external interface ConstraintSet

@kotlin.internal.InlineOnly
public inline fun ConstraintSet(): ConstraintSet {
    val o = js("({})")
    return o
}

public external interface Constraints : ConstraintSet {
    var advanced: Array<ConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints {
    val o = js("({})")
    o["advanced"] = advanced
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaStreamTrackState {
    companion object
}

public inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState get() = "live".asDynamic().unsafeCast<MediaStreamTrackState>()

public inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState get() = "ended".asDynamic().unsafeCast<MediaStreamTrackState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoFacingModeEnum {
    companion object
}

public inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum get() = "user".asDynamic().unsafeCast<VideoFacingModeEnum>()


public inline val VideoFacingModeEnum.Companion.LEFT: VideoFacingModeEnum get() = "left".asDynamic().unsafeCast<VideoFacingModeEnum>()

public inline val VideoFacingModeEnum.Companion.RIGHT: VideoFacingModeEnum get() = "right".asDynamic().unsafeCast<VideoFacingModeEnum>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoResizeModeEnum {
    companion object
}

public inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = "none".asDynamic().unsafeCast<VideoResizeModeEnum>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.VIDEOINPUT: MediaDeviceKind get() = "videoinput".asDynamic().unsafeCast<MediaDeviceKind>()
EndOfStreamError = definedExternally
fun setLiveSeekableRange(start: Double, end: Double)
fun clearLiveSeekableRange()
companion object
fun isTypeSupported(type: String): Boolean

public external abstract class SourceBuffer : EventTarget
    open val open: Boolean
    open val buffered: TimeRanges
    open val timestampOffset: Double
    open val audioTracks: AudioTrackList
    open val videoTracks: VideoTrackList
    open val textTracks: TextTrackList
    open var appendWindowStart: Double
    open var appendWindowEnd: Double
    open var onupdatestart: ((Event) -> dynamic)?
    open var onupdate: ((Event) -> dynamic)?
    open var onupdateend: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onabort: ((Event) -> dynamic)?
    fun appendBuffer(data: dynamic)
    fun abort()
    fun remove(start: Double, end: Double)

public external abstract class SourceBufferList : EventTarget
    open val length: Int
    open var onaddsourcebuffer: ((Event) -> dynamic)?
    open var onremovesourcebuffer: ((Event) -> dynamic)?

@kotlin.internal.InlineOnly
public inline operator fun SourceBufferList.get(index: Int): SourceBuffer? = asDynamic()[index]

@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface EndOfStreamError
    companion object


public inline val EndOfStreamError.Companion.DECODE: EndOfStreamError get() = "decode".asDynamic().unsafeCast<EndOfStreamError>()

@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface AppendMode
    companion object

public inline val AppendMode.Companion.SEGMENTS: AppendMode get() = "segments".asDynamic().unsafeCast<AppendMode>()

public inline val AppendMode.Companion.SEQUENCE: AppendMode get() = "sequence".asDynamic().unsafeCast<AppendMode>()
    val o = js("({})")
    o["pointerId"] = pointerId
    o["width"] = width
    o["height"] = height
    o["pressure"] = pressure
    o["tangentialPressure"] = tangentialPressure
    o["tiltX"] = tiltX
    o["tiltY"] = tiltY
    o["twist"] = twist
    o["pointerType"] = pointerType
    o["isPrimary"] = isPrimary
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [PointerEvent](https://developer.mozilla.org/en/docs/Web/API/PointerEvent) to Kotlin
 */
public external open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally) : MouseEvent {
    open val pointerId: Int
    open val width: Double
    open val height: Double
    open val pressure: Float
    open val tangentialPressure: Float
    open val tiltX: Int
    open val tiltY: Int
    open val twist: Int
    open val pointerType: String
    open val isPrimary: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!// See github.com/kotlin/dukat for details

package org.w3c.dom.svg

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin
 */
public external abstract class SVGElement : Element, ElementCSSInlineStyle, GlobalEventHandlers, SVGElementInstance {
    open val dataset: DOMStringMap
    open val ownerSVGElement: SVGSVGElement?
    open val viewportElement: Element?
    open val tabIndex: Int
    open var tabIndex: Int
    fun focus()
    fun blur()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
SVGBoundingBoxOptions {
    var fill: Boolean? /* = true */
    get() = definedExternally
    set(value) = definedExternally

    var stroke: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally

    var markers: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally

    var clipped: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["fill"] = fill
    o["stroke"] = stroke
    o["markers"] = markers
    o["clipped"] = clipped
    return o
}

/**
 * Exposes the JavaScript
[SVGGraphicsElement](https://developer.mozilla.org/en/docs/Web/API/SVGGraphicsElement) to Kotlin
*/
public external abstract class SVGGraphicsElement : SVGElement, SVGTests {
    open val transform: SVGAnimatedTransformList
    fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect
    fun getCTM(): DOMMatrix?
    fun getScreenCTM(): DOMMatrix?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin
*/
public external abstract class SVGGeometryElement : SVGGraphicsElement {
    open val pathLength: SVGAnimatedNumber
    fun isPointInFill(point: DOMPoint): Boolean
    fun isPointInStroke(point: DOMPoint): Boolean
    fun getTotalLength(): Float
    fun getPointAtLength(distance: Float): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin
*/
public external abstract class SVGLength {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_LENGTHTYPE_UNKNOWN: Short
        val SVG_LENGTHTYPE_NUMBER: Short
        val SVG_LENGTHTYPE_PERCENTAGE: Short
        val SVG_LENGTHTYPE_EMS: Short
        val SVG_LENGTHTYPE_EXS: Short
        val SVG_LENGTHTYPE_CM: Short
        val SVG_LENGTHTYPE_MM: Short
        val SVG_LENGTHTYPE_IN: Short
        val SVG_LENGTHTYPE_PT: Short
        val SVG_LENGTHTYPE_PC: Short
    }
}

/**
 * Exposes the JavaScript
[SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin
*/
public external abstract class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
}

Float\n open var valueAsString: String\n fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)\n companion object {\n    \n val SVG_ANGLETYPE_UNKNOWN: Short\n val SVG_ANGLETYPE_UNSPECIFIED: Short\n val SVG_ANGLETYPE_DEG: Short\n val SVG_ANGLETYPE_GRAD: Short\n }\n\npublic external abstract class SVGNameList {\n    open val length: Int\n    open val numberOfItems: Int\n    fun clear()\n    fun initialize(newItem: dynamic): dynamic\n    fun insertItemBefore(newItem: dynamic, index: Int): dynamic\n    fun replaceItem(newItem: dynamic, index: Int): dynamic\n    fun removeItem(index: Int): dynamic\n    fun appendItem(newItem: dynamic): dynamic\n    fun getItem(index: Int): dynamic\n}\n
@kotlin.internal.InlineOnly\npublic inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]}

@kotlin.internal.InlineOnly\npublic inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin
 */
public external abstract class SVGNumberList {\n    open val length: Int\n    open val numberOfItems: Int\n    fun clear()\n    fun initialize(newItem: SVGNumber): SVGNumber\n    fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber\n    fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber\n    fun removeItem(index: Int): SVGNumber\n    fun appendItem(newItem: SVGNumber): SVGNumber\n    fun getItem(index: Int): SVGNumber\n}\n
@kotlin.internal.InlineOnly\npublic inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]}

@kotlin.internal.InlineOnly\npublic inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin
 */
public external abstract class SVGLengthList {\n    open val length: Int\n    open val numberOfItems: Int\n    fun clear()\n    fun initialize(newItem: SVGLength): SVGLength\n    fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength\n    fun replaceItem(newItem: SVGLength, index: Int): SVGLength\n    fun removeItem(index: Int): SVGLength\n    fun appendItem(newItem: SVGLength): SVGLength\n    fun getItem(index: Int): SVGLength\n}\n
@kotlin.internal.InlineOnly\npublic inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]}

@kotlin.internal.InlineOnly\npublic inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin
 */
public external abstract class SVGAnimatedBoolean {\n    open var baseVal: Boolean\n    open val animVal: Boolean\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin
 */
public external abstract class SVGAnimatedEnumeration {\n    open var baseVal: Short\n    open val animVal: Short\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin
 */
public external abstract class SVGAnimatedInteger {\n    open var baseVal: Int\n    open val animVal: Int\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin
 */
public external abstract class SVGAnimatedNumber {\n    open var baseVal: Float\n    open val animVal: Float\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin
 */
public external abstract class SVGAnimatedLength {\n    open var baseVal: SVGLength\n    open val animVal: SVGLength\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin
 */
public external abstract class SVGAnimatedAngle {\n    open var baseVal: SVGAngle\n    open val animVal: SVGAngle\n}\n
/**
 * Exposes the JavaScript [SVGAnimatedString](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedString) to Kotlin
 */
public external abstract class SVGAnimatedString {\n    open var baseVal: String\n    open val animVal: String\n}
* Exposes the JavaScript [SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect) to Kotlin

```kotlin
public external abstract class SVGAnimatedRect {
    open val baseVal: DOMRect
    open val animVal: DOMRectReadOnly
}
```

* Exposes the JavaScript [SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin

```kotlin
public external abstract class SVGAnimatedNumberList {
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberListReadOnly
}
```

* Exposes the JavaScript [SVGAnimatedLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLengthList) to Kotlin

```kotlin
public external abstract class SVGAnimatedLengthList {
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthListReadOnly
}
```

* Exposes the JavaScript [SVGStringList](https://developer.mozilla.org/en/docs/Web/API/SVGStringList) to Kotlin

```kotlin
public external abstract class SVGStringList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun insertItemBefore(newItem: String, index: Int): String
    fun replaceItem(newItem: String, index: Int): String
    fun removeItem(index: Int): String
    fun appendItem(newItem: String): String
    fun getItem(index: Int): String
}
```

@kotlin.internal.InlineOnly
```kotlin
public inline operator fun SVGStringList.get(index: Int): String? = asDynamic()[index]
```

@kotlin.internal.InlineOnly
```kotlin
public inline operator fun SVGStringList.set(index: Int, newItem: String) { asDynamic()[index] = newItem }
```

* Exposes the JavaScript [SVGUnitTypes](https://developer.mozilla.org/en/docs/Web/API/SVGUnitTypes) to Kotlin

```kotlin
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGUnitTypes {[
    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    }
}
```

* Exposes the JavaScript [SVGTests](https://developer.mozilla.org/en/docs/Web/API/SVGTests) to Kotlin

```kotlin
public external interface SVGTests {
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
}
```

* Exposes the JavaScript [SVGZoomAndPan](https://developer.mozilla.org/en/docs/Web/API/SVGZoomAndPan) to Kotlin

```kotlin
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGZoomAndPan {
    var zoomAndPan: Short

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
    }
}
```

* Exposes the JavaScript [SVGURIReference](https://developer.mozilla.org/en/docs/Web/API/SVGURIReference) to Kotlin

```kotlin
public external interface SVGURIReference {
    val href: SVGAnimatedString
}
```

* Exposes the JavaScript [SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin

```kotlin
public external abstract class SVGSVGElement : SVGGraphicsElement, SVGFitToViewBox, WindowEventHandlers {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open var currentScale: Float
    open val currentTranslate: DOMPointReadOnly
    fun getIntersectionList(rect: DOMRectReadOnly): NodeList
    fun getEnclosureList(rect: DOMRectReadOnly): NodeList
    fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun deselectAll()
    fun createSVGNumber(): SVGNumber
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGMatrix(): DOMMatrix
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun getElementById(elementId: String): Element
    fun suspendRedraw(maxWaitMilliseconds: Int): Int
    fun unsuspendRedraw(suspendHandleID: Int)
    fun unsuspendRedrawAll()
    fun forceRedraw()

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE:
Open Source Used In DNAC Platform Arica 8241

val SVG_ZOOMANDPAN_MAGNIFY: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript
 * [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin
 */
public external abstract class SVGGElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGUnknownElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 * [SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin
 */
public external abstract class SVGDefsElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 * [SVGDescElement](https://developer.mozilla.org/en/docs/Web/API/SVGDescElement) to Kotlin
 */
public external abstract class SVGDescElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
/**
 * Exposes the JavaScript
 [SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin

 */
public external abstract class SVGMetadataElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [SVGTitl
DOCUMENT_POSITION_DISCONNECTED: Short
DOCUMENT_POSITION_PRECEDING: Short
DOCUMENT_POSITION_FOLLOWING: Short
DOCUMENT_POSITION_CONTAINS: Short
DOCUMENT_POSITION_CONTAINED_BY: Short
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public open class SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public interface SVGElementInstance {
    val correspondingElement: SVGElement?
        get() = definedExternally
    val correspondingUseElement: SVGUseElement?
        get() = definedExternally
}

public open class ShadowAnimation(source: dynamic, newTarget: dynamic) {
    open val sourceAnimation: dynamic
}

/**
 * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin
 */
public abstract class SVGSwitchElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public interface GetSVGDocument {
    fun getSVGDocument(): Document
}

/**
 * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
 */
public abstract class SVGStyleElement : SVGElement, LinkStyle {
    open var type: String
    open var media: String
    open var title: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin
 */
public abstract class SVGTransform {
    open val type: Short
    open val matrix: DOMMatrix
    open val angle: Float
    fun setMatrix(matrix: DOMMatrixReadOnly)
    fun setTranslate(tx: Float, ty: Float)
    fun setScale(sx: Float, sy: Float)
    fun setRotate(angle: Float, cx: Float, cy: Float)
    fun setSkewX(angle: Float)
    fun setSkewY(angle: Float)

    companion object {
        val SVG_TRANSFORM_UNKNOWN: Short
        val SVG_TRANSFORM_MATRIX: Short
        val SVG_TRANSFORM_TRANSLATE: Short
        val SVG_TRANSFORM_SCALE: Short
        val SVG_TRANSFORM_ROTATE: Short
    }
}
Open Source Used In DNAC Platform Arica 8244

```kotlin
/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
public external abstract class SVGTransformList {
    open val length: Int
    open val numberOfItems: Int

    fun clear()
    fun initialize(newItem: SVGTransform): SVGTransform
    fun insertItemBefore(newItem: SVGTransform, index: Int): SVGTransform
    fun replaceItem(newItem: SVGTransform, index: Int): SVGTransform
    fun removeItem(index: Int): SVGTransform
    fun appendItem(newItem: SVGTransform): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
    fun consolidate(): SVGTransform?
    fun getItem(index: Int): SVGTransform
}

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.get(index: Int): SVGTransform? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGAnimatedTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedTransformList) to Kotlin
 */
public external abstract class SVGAnimatedTransformList {
    open val baseVal: SVGTransformList
    open val animVal: SVGTransformList
}

/**
 * Exposes the JavaScript [SVGPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGPreserveAspectRatio) to Kotlin
 */
public external abstract class SVGPreserveAspectRatio {
    open var align: Short
    open var meetOrSlice: Short

    companion object {
        val SVG_PRESERVEASPECTRATIO_UNKNOWN: Short
        val SVG_PRESERVEASPECTRATIO_NONE: Short
        val SVG_PRESERVEASPECTRATIO_XMINYMIN: Short
        val SVG_PRESERVEASPECTRATIO_XMIDYMIN: Short
        val SVG_PRESERVEASPECTRATIO_XMAXYMIN: Short
        val SVG_PRESERVEASPECTRATIO_XMINYMID: Short
        val SVG_PRESERVEASPECTRATIO_XMIDYMID: Short
        val SVG_PRESERVEASPECTRATIO_XMAXYMID: Short
        val SVG_PRESERVEASPECTRATIO_XMINYMAX: Short
        val SVG_PRESERVEASPECTRATIO_XMIDYMAX: Short
        val SVG_PRESERVEASPECTRATIO_XMAXYMAX: Short
        val SVG_MEETORSLICE_UNKNOWN: Short
        val SVG_MEETORSLICE_MEET: Short
        val SVG_MEETORSLICE_SLICE: Short
    }
}

/**
 * Exposes the JavaScript [SVGAnimatedPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPreserveAspectRatio) to Kotlin
 */
public external abstract class SVGAnimatedPreserveAspectRatio {
    open val baseVal: SVGPreserveAspectRatio
    open val animVal: SVGPreserveAspectRatio
}

/**
 * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
 */
public external abstract class SVGPathElement : SVGGeometryElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

[SVGAnimatedTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedTransformList) to Kotlin

[SVGPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGPreserveAspectRatio) to Kotlin

[SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear(): Unit
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
    fun removeItem(index: Int): DOMPoint
    fun appendItem(newItem: DOMPoint): DOMPoint
    fun getItem(index: Int): DOMPoint
}

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()\[index\]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()\[index\] = newItem }

public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGTextContentElement : SVGGraphicsElement {
    open val textLength: SVGAnimatedLength
    open val lengthAdjust: SVGAnimatedEnumeration
    fun getNumberOfChars(): Int
    fun getComputedTextLength(): Float
    fun getSubStringLength(charnum: Int, nchars: Int): Float
    fun getStartPositionOfChar(charnum: Int): DOMPoint
    fun getEndPositionOfChar(charnum: Int): DOMPoint
    fun getExtentOfChar(charnum: Int): DOMRect
    fun getRotationOfChar(charnum: Int): Float
    fun getCharNumAtPosition(point: DOMPoint): Int
    fun selectSubString(charnum: Int, nchars: Int, charnum: Int, nchars: Int)

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
    }
}
The image appears to contain a block of code written in a programming language, possibly for exposure of JavaScript elements (SVGTextPathElement, SVGImageElement, SVGForeignObjectElement) to Kotlin. The text includes various properties and methods related to these elements, such as `startOffset`, `method`, `spacing`, `elementNode`, and `attributeNode`.

The code also includes a companion object with various `Short` values representing different states and functionalities of the elements, such as `TEXTPATH_METHODTYPE_UNKNOWN`, `TEXTPATH_METHODTYPE_ALIGN`, `TEXTPATH_SPACINGTYPE_UNKNOWN`, `TEXTPATH_SPACINGTYPE_AUTO`, `LENGTHADJUST_UNKNOWN`, etc.

This block of code seems to be part of a larger program that deals with SVG elements and their properties, possibly for dynamic web content generation or interaction with the user interface.
val SVG_SPREADMETHOD_PAD: Short
val SVG_SPREADMETHOD_REPEAT: Short
val SVG_UNIT_TYPE_UNKNOWN: Short
val SVG_UNIT_TYPE_USERSPACEONUSE: Short
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGRadialGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGRadialGradientElement) to Kotlin
 * *\public external abstract class SVGRadialGradientElement : SVGGradientElement {\n open val cx: SVGAnimatedLength open val cy: SVGAnimatedLength open val r: SVGAnimatedLength open val fx: SVGAnimatedLength open val fy: SVGAnimatedLength companion object {\n val SVG_SPREADMETHOD_UNKNOWN: Short
val SVG_SPREADMETHOD_PAD: Short
val SVG_SPREADMETHOD_REPEAT: Short
val SVG_UNIT_TYPE_UNKNOWN: Short
val SVG_UNIT_TYPE_USERSPACEONUSE: Short
val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
} */

public external abstract class SVGRadialGradientElement : SVGGradientElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength
    open val fx: SVGAnimatedLength
    open val fy: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGMeshGradientElement : SVGGradientElement {
    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGMeshrowElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
public external abstract class SVGStopElement : SVGElement {
    open val offset: SVGAnimatedNumber

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGStopElement] to Kotlin */
public external abstract class SVGStopElement : SVGElement {
    open val offset: SVGAnimatedNumber

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGPatternElement] to Kotlin */
public external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {
    open val patternUnits: SVGAnimatedEnumeration
    open val patternContentUnits: SVGAnimatedEnumeration
    open val patternTransform: SVGAnimatedTransformList
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGHatchElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external abstract class SVG HatchpathElement : SVGElement {

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGCursorElement](https://developer.mozilla.org/en/docs/Web/API/SVGCursorElement) to Kotlin
 */
public external abstract class SVGCursorElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin
 */
public external abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLOrSVGScriptElement {
    open var type: String
    open var crossOrigin: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGAElement](https://developer.mozilla.org/en/docs/Web/API/SVGAElement) to Kotlin
 */
public external abstract class SVGAElement : SVGGraphicsElement, SVGURIReference {
    open val target: SVGAnimatedString
    open val download: SVGAnimatedString
    open val rel: SVGAnimatedString
    open val relList: SVGAnimatedString
    open val hreflang: SVGAnimatedString
    open val type: SVGAnimatedString

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Open Source Used In DNAC Platform Arica 8253
var onloadend: ((Event) -> dynamic)?
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsDataURL(blob: Blob): String
    fun abort(): Unit

    companion object {
        val EMPTY: Short
        val LOADING: Short
        val DONE: Short
    }

/**
 * Exposes the JavaScript [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}

/**
 * Exposes the JavaScript [Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin
 */
public external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    var onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: Array<out Int>
    open val timestamp: Number
    open val renotify: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun close()

    companion object {
        val permission: NotificationPermission
        val maxActions: Int
        fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>
    }
}

public external interface NotificationOptions {
    var dir: NotificationDirection? /* = NotificationDirection.AUTO */
    get() = definedExternally
        set(value) = definedExternally
    var lang: String?
        get() = definedExternally
        set(value) = definedExternally
    var body: String?
        get() = definedExternally
        set(value) = definedExternally
    var tag: String?
        get() = definedExternally
        set(value) = definedExternally
    var image: String?
        get() = definedExternally
        set(value) = definedExternally
    var icon: String?
        get() = definedExternally
        set(value) = definedExternally
    var badge: String?
        get() = definedExternally
        set(value) = definedExternally
    var sound: String?
        get() = definedExternally
        set(value) = definedExternally
    var vibrate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var timestamp: Number?
        get() = definedExternally
        set(value) = definedExternally
    var renotify: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var silent: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var noscreen: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var requireInteraction: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var sticky: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var data: Any?
        get() = definedExternally
        set(value) = definedExternally
    var actions: Array<NotificationAction>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly

false, data: Any? = null, actions: Array<NotificationAction> = arrayOf()): NotificationOptions
{
    var o = js("({})")
    o["dir"] = dir
    o["lang"] = lang
    o["body"] = body
    o["tag"] = tag
    o["image"] = image
    o["icon"] = icon
    o["badge"] = badge
    o["sound"] = sound
    o["vibrate"] = vibrate
    o["timestamp"] = timestamp
    o["renotify"] = renotify
    o["silent"] = silent
    o["noscreen"] = noscreen
    o["requireInteraction"] = requireInteraction
    o["sticky"] = sticky
    o["data"] = data
    o["actions"] = actions
    return o
}

public external interface NotificationAction {
    var action: String?
        get() = definedExternally
        set(value) = definedExternally
    var title: String?
        get() = definedExternally
        set(value) = definedExternally
    var icon: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
{
    var o = js("({})")
    o["action"] = action
    o["title"] = title
    o["icon"] = icon
    return o
}

public external interface GetNotificationOptions {
    var tag: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetNotificationOptions(tag: String? = ""): GetNotificationOptions
{
    var o = js("({})")
    o["tag"] = tag
    return o
}

/**
 * Exposes the JavaScript [NotificationEvent](https://developer.mozilla.org/en/docs/Web/API/NotificationEvent) to Kotlin
 */
public external open class NotificationEvent(type: String, eventInitDict: NotificationEventInit) : ExtendableEvent
{
    open val notification: Notification
    open val action: String

    companion object
    {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface NotificationEventInit : ExtendableEventInit
    {
        var notification: Notification?
            get() = definedExternally
            set(value) = definedExternally
        var action: String? /* = "" */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    {
        var o = js("({})")
        o["notification"] = notification
        o["action"] = action
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /* please, don't implement this interface! */
    @Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
    public external interface NotificationPermission
    {
        companion object
    }

    public inline val NotificationPermission.Companion.DEFAULT: NotificationPermission
        get() = "default".asDynamic().unsafeCast<NotificationPermission>()
    public inline val NotificationPermission.Companion.DENIED: NotificationPermission
        get() = "denied".asDynamic().unsafeCast<NotificationPermission>()
    public inline val NotificationPermission.Companion.GRANTED: NotificationPermission
        get() = "granted".asDynamic().unsafeCast<NotificationPermission>()

    /* please, don't implement this interface! */
    @Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
    public external interface NotificationDirection
    {
        companion object
    }

    public inline val NotificationDirection.Companion.AUTO: NotificationDirection
        get() = "auto".asDynamic().unsafeCast<NotificationDirection>()
    public inline val NotificationDirection.Companion.LTR: NotificationDirection
        get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()
    public inline val NotificationDirection.CompanionRTL: NotificationDirection
        get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()

    /* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. */
    /* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */
    "*/
org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.*/!

/**
 * Exposes the JavaScript ServiceWorker to Kotlin
 */
public external abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open val scriptURL: String
    open val state: ServiceWorkerState
    open var onstatechange: ((Event) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript ServiceWorkerRegistration to Kotlin
 */
public external abstract class ServiceWorkerRegistration : EventTarget {
    open val installing: ServiceWorker?
    open val waiting: ServiceWorker?
    open val active: ServiceWorker?
    open val scope: String
    open var onupdatefound: ((Event) -> dynamic)?
    val APISpace: dynamic
    fun update(): Promise<Unit>
    fun unregister(): Promise<Boolean>
    fun showNotification(title: String, options: NotificationOptions = definedExternally): Promise<{}>
    fun method(): Promise<dynamic>
}

/**
 * Exposes the JavaScript ServiceWorkerContainer to Kotlin
 */
public external abstract class ServiceWorkerContainer : EventTarget {
    open val controller: ServiceWorker?
    open val ready: Promise<ServiceWorkerRegistration>
    open var oncontrollerchange: ((Event) -> dynamic)?
    fun register(scriptURL: String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>
    fun getRegistration(clientURL: String = definedExternally): Promise<any>?
    fun startMessages()
}

public external interface RegistrationOptions {
    var scope: String?
        get() = definedExternally
        set(value) = definedExternally
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {
    val o = js("({})")
    o["scope"] = scope
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript ServiceWorkerGlobalScope to Kotlin
 */
public external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {
    open val clients: Clients
    open val registration: ServiceWorkerRegistration
    open var oninstall: ((Event) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}
var onactivate: ((Event) -> dynamic)?
open var onfetch: ((FetchEvent) -> dynamic)?
on open var onforeignfetch: ((Event) -> dynamic)?
on open var onmessage: ((MessageEvent) -> dynamic)?
on open var onnotificationclick: ((NotificationEvent) -> dynamic)?
on open var onnotificationclose: ((NotificationEvent) -> dynamic)?
on open var onfunctionalevent: ((Event) -> dynamic)?
fun skipWaiting():
Promise<Unit>()

/**
 * Exposes the JavaScript [Client](https://developer.mozilla.org/en/docs/Web/API/Client) to Kotlin
 */
public external abstract class Client :
UnionClientOrMessagePortOrServiceWorker {
open val url: String
open val frameType: FrameType
open val id: String
fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript [WindowClient](https://developer.mozilla.org/en/docs/Web/API/WindowClient) to Kotlin
 */
public external abstract class WindowClient : Client {
open val visibilityState: dynamic
open val focused: Boolean
fun focus(): Promise<WindowClient>
fun navigate(url: String):
}

/**
 * Exposes the JavaScript [Clients](https://developer.mozilla.org/en/docs/Web/API/Clients) to Kotlin
 */
public external abstract class Clients {
fun get(id: String): Promise<Any?>
fun matchAll(options: ClientQueryOptions =
definedExternally): Promise<Array<Client>>
fun openWindow(url: String):
}

public external interface ClientQueryOptions {
var includeUncontrolled: Boolean? /* = false */
var type: ClientType? /* = ClientType.WINDOW */
}

@kotlin.internal.InlineOnly
public inline fun ClientQueryOptions(includeUncontrolled: Boolean? = false, type: ClientType? = ClientType.WINDOW): ClientQueryOptions {
val o = js("({})")
   o["includeUncontrolled"] = includeUncontrolled
   o["type"] = type
   return o
}

/**
 * Exposes the JavaScript [ExtendableEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableEvent) to Kotlin
 */
public external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {
fun waitUntil(f: Promise<Any?>)

companion object {
val NONE: Short
val CAPTURING_PHASE: Short
val AT_TARGET: Short
val BUBBLING_PHASE: Short
}

public external interface ExtendableEventInit : EventInit

@kotlin.internal.InlineOnly
val o = js("({})")
   o["bubbles"] = bubbles
   o["cancelable"] = cancelable
   o["composed"] = composed
   return o
}

/**
 * Exposes the JavaScript [InstallEvent](https://developer.mozilla.org/en/docs/Web/API/InstallEvent) to Kotlin
 */
public external open class InstallEvent(type: String, eventInitDict: InstallEventInit = definedExternally) : ExtendableEvent {
fun registerForeignFetch(options: ForeignFetchOptions)

companion object {
val NONE: Short
val CAPTURING_PHASE: Short
val AT_TARGET: Short
val BUBBLING_PHASE: Short
}

public external interface ForeignFetchOptions {
var scopes: Array<String>?
var origins: Array<String>?
}

@kotlin.internal.InlineOnly
public inline fun ForeignFetchOptions(scopes: Array<String>?, origins: Array<String>?): ForeignFetchOptions {
val o = js("({})")
   o["scopes"] = scopes
   o["origins"] = origins
   return o
}

/**
 * Exposes the JavaScript [FetchEvent](https://developer.mozilla.org/en/docs/Web/API/FetchEvent) to Kotlin
 */
public external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {
fun respondWith(r: Promise<Response>)

companion object {
val NONE: Short
val CAPTURING_PHASE: Short
val AT_TARGET: Short
val BUBBLING_PHASE: Short
}

public external interface FetchEventInit : ExtendableEventInit {
var request: Request
var credentials: String
var method: String
var redirect: Boolean
var headers: Array<mutable.Map<String, String>>
}

@kotlin.internal.InlineOnly
val o = js("({})")

```kotlin
o["request"] = request
    o["clientId"] = clientId
    o["isReload"] = isReload
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external open class ForeignFetchEvent(type: String, eventInitDict: ForeignFetchEventInit) : ExtendableEvent {
    open val request: Request
    open val origin: String
    fun respondWith(r: Promise<ForeignFetchResponse>)
}

public external interface ForeignFetchEventInit : ExtendableEventInit {
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["request"] = request
    o["origin"] = origin
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin
 * in an external interface ExtendableMessageEventInit : ExtendableEvent {
 *    open val request: Promise<ForeignFetchResponse>
 *    open val origin: String
 *    companion object {
 *        val NONE: Short
 *        val CAPTURING_PHASE: Short
 *        val AT_TARGET: Short
 *        val BUBBLING_PHASE: Short
 *    }
 *}
 * @kotlin.internal.InlineOnly
 * public inline fun ExtendableMessageEventInit(request: Promise<ForeignFetchResponse>, origin: String? = undefined): ExtendableMessageEventInit {
 *    val o = js("({})")
 *    o["request"] = request
 *    o["origin"] = origin
 *    return o
 *}
 * val messagePorts: Array<MessagePort> = ...
 * fun receive(message: Message) {
 *    messagePorts.forEach { port -> port.onmessage?.invoke(message) }
 *}
 */

class MyCustomEventListener {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    fun register(messagePort: MessagePort) {
        messagePort.onmessage = { message -> processMessage(message) }
    }

    private fun processMessage(message: Message) {
        // Process the message
    }
}
```
get() = definedExternally
set(value) = definedExternally

    val o = js("{}")
o["ignoreSearch"] = ignoreSearch
    o["ignoreMethod"] = ignoreMethod
    o["ignoreVary"] = ignoreVary
    o["cacheName"] = cacheName
    return o
}

    val o = js("{}")
o["type"] = type
    o["request"] = request
    o["response"] = response
    return o
}

interface CacheStorage {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun has(cacheName: String): Promise<Boolean>
    fun open(cacheName: String): Promise<Cache>
    fun delete(cacheName: String): Promise<Boolean>
    fun keys(): Promise<Array<String>>
}

interface FunctionalEvent {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

interface UnionMessagePortOrServiceWorker {
    companion object
}

interface UnionClientOrMessagePortOrServiceWorker {
    companion object
}

interface ServiceWorkerState {
    companion object
}

val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState = "installing".asDynamic().unsafeCast<ServiceWorkerState>()
val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState = "activating".asDynamic().unsafeCast<ServiceWorkerState>()
val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState = "activated".asDynamic().unsafeCast<ServiceWorkerState>()
val ServiceWorkerState.Companion.REDUNDANT: ServiceWorkerState = "redundant".asDynamic().unsafeCast<ServiceWorkerState>()

interface FrameType {
    companion object
}

val FrameType.Companion.AUXILIARY: FrameType = "auxiliary".asDynamic().unsafeCast<FrameType>()
val FrameType.Companion.TOP_LEVEL: FrameType = "top-level".asDynamic().unsafeCast<FrameType>()
val FrameType.Companion.NESTED: FrameType = "nested".asDynamic().unsafeCast<FrameType>()
val FrameType.Companion.NONE: FrameType = "none".asDynamic().unsafeCast<FrameType>()

interface ClientType {
    companion object
}

val ClientType.Companion.WINDOW: ClientType = "window".asDynamic().unsafeCast<ClientType>()
val ClientType.Companion.WORKER: ClientType = "worker".asDynamic().unsafeCast<ClientType>()
val ClientType.Companion.SHAREDWORKER: ClientType = "sharedworker".asDynamic().unsafeCast<ClientType>()
val ClientType.Companion.ALL: ClientType = "all".asDynamic().unsafeCast<ClientType>()

package org.w3c.xhr

import kotlin.js.*
import

org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*

/**
 * Exposes the JavaScript
 [XMLHttpRequestEventTarget](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequestEventTarget) to
Kotlin
 */
public external abstract class XMLHttpRequestEventTarget : EventTarget {
    open var onloadstart: ((ProgressEvent) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onabort: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onload: ((Event) -> dynamic)?
    open var ontimeout: ((Event) -> dynamic)?
    open var onloadend: ((Event) -> dynamic)?
}

public external abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget

/**
 * Exposes the JavaScript
 [XMLHttpRequest](https://developer.mozilla.org/en/docs/Web/API/XMLHttpRequest) to Kotlin
 */
public external open class XMLHttpRequest : XMLHttpRequestEventTarget {
    var onreadystatechange: ((Event) -> dynamic)?
    open val readyState: Short?
    var timeout: Int
    var withCredentials: Boolean
    open val upload: XMLHttpRequestUpload
    open val responseURL: String
    open val status: Short?
    open val statusText: String
    var responseType: XMLHttpRequestResponseType
    open val response: Any?
    open val responseText: String
    open val responseXML: Document?
    fun open(method: String, url: String)
    fun open(method: String, url: String, async: Boolean, username: String? = definedExternally, password: String? =
definedExternally)
    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String?
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Short
        val OPENED: Short
        val HEADERS_RECEIVED: Short
        val LOADING: Short
        val DONE: Short
    }
}

/**
 * Exposes the JavaScript
 [FormData](https://developer.mozilla.org/en/docs/Web/API/FormData) to Kotlin
 */
public external open class FormData(form: HTMLFormElement = definedExternally) {
    fun append(name: String, value: String)
    fun append(name: String, value: Blob, filename: String = definedExternally)
    fun delete(name: String)
    fun get(name: String): dynamic
    fun getAll(name: String): Array<dynamic>
    fun has(name: String): Boolean
    fun set(name: String, value: String)
    fun set(name: String, value: Blob, filename: String = definedExternally)
}

/**
 * Exposes the JavaScript
 [ProgressEvent](https://developer.mozilla.org/en/docs/Web/API/ProgressEvent) to Kotlin
 */
public external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) : Event {
    open val lengthComputable: Boolean
    open val loaded: Number
    open val total: Number

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ProgressEventInit : EventInit {
    var lengthComputable: Boolean? /* = false */
    var loaded: Number? /* = 0 */
    var total: Number? /* = 0 */
}

@kotlin.internal.InlineOnly
    return ProgressEventInit {
        lengthComputable = lengthComputable
        loaded = loaded
        total = total
        bubbles = bubbles
        cancelable = cancelable
        composed = composed
    }
}

/* please, don’t implement this
interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface XMLHttpRequestResponseType {
    companion object
}

public inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() = "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() =
public inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = "blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = "document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = "json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = "text".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationRetention.BINARY
import kotlin.annotation.AnnotationRetention.SOURCE
import kotlin.annotation.AnnotationTarget.*
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind
import kotlin.reflect.KClass

/**
 * Signals that the annotated annotation class is a marker of an experimental API.
 * Any declaration annotated with that marker is considered an experimental declaration
 * and its call sites should accept the experimental aspect of it either by using [UseExperimental],
 * or by being annotated with that marker themselves, effectively causing further propagation of that experimental aspect.
 * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
 */
@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Deprecated("Please use RequiresOptIn instead.")
public annotation class Experimental(val level: Level = Level.ERROR) {
    /**
     * Severity of the diagnostic that should be reported on usages of experimental API which did not explicitly accept the experimental aspect
     * of that API either by using [UseExperimental] or by being annotated with the corresponding marker annotation.
     * If a declaration is annotated with [UseExperimental], its usages are **not** required to opt-in to that experimental API.
     * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
     */
    public enum class Level {
        /** Specifies that a warning should be reported on incorrect usages of this experimental API. */
        WARNING,
        /** Specifies that an error should be reported on incorrect usages of this experimental API. */
        ERROR,
    }
}

/**
 * Allows to use experimental API denoted by the given markers in the annotated file, declaration, or expression.
 * If a declaration is annotated with [UseExperimental], its usages are **not** required to opt-in to that experimental API.
 * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
 */
@Target(
    CLASS,
    PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION,
    PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
    TYPEALIAS)
@Retention(SOURCE)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Deprecated("Please use OptIn instead.", ReplaceWith("OptIn(*markerClass)", "kotlin.OptIn"))
public annotation class UseExperimental(
    vararg val markerClass: KClass<out Annotation>
)

@Target(CLASS, PROPERTY, CONSTRUCTOR, FUNCTION,
    PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
    TYPEALIAS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Deprecated("Please use OptIn instead.", ReplaceWith("OptIn(*markerClass)", "kotlin.OptIn"))
@Suppress("DEPRECATION")
public annotation class WasExperimental(
    vararg val markerClass: KClass<out Annotation>
)

@Target(CLASS, PROPERTY, CONSTRUCTOR, FUNCTION,
    PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
    TYPEALIAS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@OptIn(CLASS)
ExperimentalStdlibApi

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationTarget.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Allows to infer generic type arguments of a function from
 * the calls in the annotated function parameter of that function.
 * When this annotation is placed on a generic function parameter of a function,
 * it enables to infer the type arguments of that generic function from
 * the lambda body passed to that parameter.
 * The calls that affect inference are either members of the receiver type of
 * an annotated function parameter or extensions for that type.
 * The extensions must be themselves annotated with `@BuilderInference`.
 * Example: we declare
 * ```
 * fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T>
 * ```
 * and use it like
 * ```
 * val result = sequence { yield("result") }
 * ```
 * Here the type argument of the resulting sequence is inferred to `String` from
 * the argument of the [SequenceScope.yield] function, that is called inside the lambda passed to
 * [sequence].
 * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.
 */
@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
@ExperimentalTypeInference
public annotation class BuilderInference

/**
 * Enables overload selection based on the type of the value
 * returned from lambda argument.
 * When two or more function overloads have otherwise the same parameter
 * lists that differ only in the return type of a functional parameter,
 * this annotation enables overload selection by the type
 * of the value returned from the lambda function passed to this functional parameter.
 * Example:
 * ```
 * @OverloadResolutionByLambdaReturnType
 * fun create(intProducer: () -> Int): Int
 *
 * fun create(doubleProducer: () -> Double): Double
 *
 * val newValue = create { 3.14 }
 * ```
 * The annotation being applied to one of overloads allows to resolve this ambiguity by analyzing what value is returned from
 * the lambda function.
 * This annotation is also used to discriminate the annotated overloads in case if overload
 * selection still cannot choose one of them even taking in account the result of lambda parameter analysis.
 * In that case a warning is reported.
 * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.
 */
@Target(FUNCTION)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.4")
@Experimental
@RequiresOptIn
@MustBeDocumented
@Target(CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER)
@TypeAlias
@Retention(AnnotationRetention.BINARY)
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPLIER_VERSION)
public annotation class ExperimentalMultiplatform

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationTarget.*
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind

/**
 * The experimental multiplatform support API marker.
 * Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by
 * annotating that usage with the [OptIn] annotation, e.g. `@OptIn(ExperimentalMultiplatform::class)`,
 * or by using the compiler argument `-Xopt-in=kotlin.ExperimentalMultiplatform`.
 */
@Suppress("DEPRECATION")
@MustBeDocumented
@Target(CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER)
@TypeAlias
@Retention(AnnotationRetention.BINARY)
@RequireKotlin("1.3")
public annotation class Experimental
corresponding actual class on a platform, * the annotation entries where it's used are simply erased when compiling code on that platform. * Note: this annotation is experimental, see [ExperimentalMultiplatform] on how to opt-in for it.

```kotlin
@Target(ANNOTATION_CLASS)
@Retention(AnnotationRetention.BINARY)
@ExperimentalMultiplatform
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@public annotation class OptionalAnnotation
```

- Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. *
- package kotlin

```kotlin
@ExperimentalMultiplatform
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class RequiresOptIn {
    
    val message: String = "",
    level: Level = Level.ERROR
}
```

- The default message is: "This declaration is experimental and its usage should be marked with 'Marker' or '@OptIn(Marker::class)'", where 'Marker' is the opt-in requirement marker. *

```kotlin
@Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE)
@Retention(SOURCE)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@public annotation class OptIn {
    vararg val markerClass: KClass<out Annotation>
}
```

- This class requires opt-in itself and can only be used with the compiler argument `-Xopt-in=kotlin.RequiresOptIn`

```kotlin
package kotlin.collections
```

- Provides a skeletal implementation of the read-only `Collection` interface.

```kotlin
@SinceKotlin("1.1")
public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }

    override fun iterator(): Iterator<E> = copyToArrayImpl(this)

    /**
     * Returns new array of type `Array<Any?>` with the elements of this collection.
     */
    @JsName("toArray")
    protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

    /**
     * Fills the provided [array] or creates new array of the same type and fills it with the elements of this collection.
     */
    override fun iterator(): Iterator<E> = copyToArrayImpl(this, array)
```

- A collection is covariant in its element type.

```kotlin
package kotlin
```

- A collection is covariant in its element type.

```
@Retention(SOURCE)
@SinceKotlin("1.1")
@public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }

    override fun iterator(): Iterator<E> = copyToArrayImpl(this)

    /**
     * Returns new array of type `Array<Any?>` with the elements of this collection.
     */
    @JsName("toArray")
    protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

    /**
     * Fills the provided [array] or creates new array of the same type and fills it with the elements of this collection.
     */
    override fun iterator(): Iterator<E> = copyToArrayImpl(this, array)
```

- A collection is covariant in its element type.

```
@Retention(SOURCE)
@SinceKotlin("1.1")
@public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }

    override fun iterator(): Iterator<E> = copyToArrayImpl(this)

    /**
     * Returns new array of type `Array<Any?>` with the elements of this collection.
     */
    @JsName("toArray")
    protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

    /**
     * Fills the provided [array] or creates new array of the same type and fills it with the elements of this collection.
     */
    override fun iterator(): Iterator<E> = copyToArrayImpl(this, array)
```

- A collection is covariant in its element type.
return
        val newCapacity = newCapacity(elementData.size, minCapacity)
        copyElements(newCapacity)
        /* * Creates a new array with the specified [newCapacity] size and copies elements in the [elementData] array to it */
        private fun copyElements(newCapacity: Int) { val newElements = arrayOfNulls<Any?>(newCapacity)
            elementData.copyInto(newElements, 0, head, elementData.size)
            elementData.copyInto(newElements, elementData.size - head, 0, head)
            head = 0
            elementData = newElements
        }

        /**
         * Creates a new array with the specified [newCapacity] size and
         * copies elements in the [elementData] array to it.
         */
        private fun copyInto(newCapacity: Int) {
            val newElements = arrayOfNulls<Any?>(newCapacity)
            elementData.copyInto(newElements, 0, head, elementData.size)
            elementData.copyInto(newElements, elementData.size - head, 0, head)
            head = 0
            elementData = newElements
        }

        /**
         * Creates a new array with the specified [newCapacity] size and
         * copies elements in the [elementData] array to it.
         */
        private fun copyInto(elementData: ArrayAny, head: Int, lastIndex: Int, size: Int) {
            val newCapacity = newCapacity(elementData.size, minCapacity)
            copyElements(newCapacity)
            if (index >= elementData.size) index = elementData.size
            if (index < 0) index = elementData.size - index
            /* * Returns the first element, or throws [NoSuchElementException] if this deque is empty.
            */
            if (isEmpty()) throw NoSuchElementException("ArrayDeque is empty.")
            /* * Returns the first element, or throws [NoSuchElementException] if this deque is empty.
            */
            val element = internalGet(head)
            elementData[head] = null
            head = incremented(head)
            size -= 1
            return element
        }

        /**
         * Removes the last element from this deque and returns that removed element, or throws [NoSuchElementException]
         * if this deque is empty.
         */
        public fun removeLast(): E {
            if (isEmpty()) throw NoSuchElementException("ArrayDeque is empty.")
            val internalLastIndex = internalIndex(lastIndex)
            val element = internalGet(internalLastIndex)
            elementData[internalLastIndex] = null
            size -= 1
            return element
        }

        /**
         * Removes the last element from this deque and returns that removed element, or returns `null` if this deque is empty.
         */
        public fun removeLastOrNull(): E? = if (isEmpty()) null else removeLast()
tail
  // -> shift all elements in the back of the array
  if (index < (size + 1) shr 1) {
  // in case the succeeding elements are shifted:
  val shiftToFront = tail + elementsSize - elementData.size
  if (shiftToFront >= elementsToShift) { // head < tail, insertion leads to head >= tail
    elementData.copyInto(elementData, head + 1, shiftedInternalIndex + 1)
  } else { // head > tail
    elementData.copyInto(elementData, head - 1, head, elementData.size)
  }
  // head can’t be zero
  elementData.copyInto(elementData, 0, 1, decrementedInternalIndex + 1)
}

private fun copyCollectionElements(internalIndex: Int, elements: Collection<E>) { //...
@Suppress("UNCHECKED_CAST")
if (predicate(element as E))

            elementData[newTail++] = element
else

                modified = true
            }

            newTail = positiveMod(newTail)

            for (index in 0 until tail) {
                val element = elementData[index]

                elementData[index] = null

                @Suppress("UNCHECKED_CAST")
                if (predicate(element as E)) {
                    elementData.fill(null, head, tail)
                } else if (isNotEmpty()) {
                    elementData.fill(null, head, elementData.size)
                }
            }

        }
        if (modified)
            size = negativeMod(newTail - head)

        return modified
    }

    public override fun clear() {
        val tail = internalIndex(size)
        if (head < tail) {
            elementData.fill(null, head, tail)
        } else if (isNotEmpty()) {
            elementData.fill(null, head, elementData.size)
            elementData.fill(null, 0, tail)
        }
        head = 0
        size = 0
    }

    internal companion object {
        private val emptyElementData = emptyArray<Any?>()
        private const val maxArraySize = Int.MAX_VALUE - 8
        private const val defaultMinCapacity = 10

        internal fun newCapacity(oldCapacity: Int, minCapacity: Int): Int {
            // overflow-conscious
            var newCapacity = oldCapacity + (oldCapacity shr 1)
            if (newCapacity - minCapacity < 0)
                newCapacity = minCapacity
            if (newCapacity - maxArraySize > 0)
                newCapacity = if (minCapacity > maxArraySize) Int.MAX_VALUE else maxArraySize
            return newCapacity
        }
    }

    // For testing only
    internal fun internalStructure(structure: (head: Int, elements: Array<Any?>) -> Unit) {
        val tail = internalIndex(size)

        if (isEmpty()) {
            structure(head, emptyArray())
            return
        }

        val elements = arrayOfNulls<Any?>(size)
        if (head < tail) {
            elementData.copyInto(elements, startIndex = head, endIndex = tail)
            structure(head, elements)
        } else {
            elementData.copyInto(elements, startIndex = head)
            elementData.copyInto(elements, elementData.size - head, startIndex = 0, endIndex = tail)
            structure(head - elementData.size, elements)
        }
    }
}
ShortArray  && v2 is ShortArray  -> if (!v1.contentEquals(v2)) return false\n v1 is IntArray  && v2 is IntArray  -> if (!v1.contentEquals(v2)) return false\n v1 is LongArray  && v2 is LongArray  -> if (!v1.contentEquals(v2)) return false\n v1 is FloatArray  && v2 is FloatArray  -> if (!v1.contentEquals(v2)) return false\n v1 is DoubleArray  && v2 is DoubleArray  -> if (!v1.contentEquals(v2)) return false\n v1 is CharArray  && v2 is CharArray  -> if (!v1.contentEquals(v2)) return false\n v1 is BooleanArray  && v2 is BooleanArray  -> if (!v1.contentEquals(v2)) return false\n v1 is UByteArray  && v2 is UByteArray  -> if (!v1.contentEquals(v2)) return false\n v1 is UShortArray  && v2 is UShortArray  -> if (!v1.contentEquals(v2)) return false\n v1 is UIntArray  && v2 is UIntArray  -> if (!v1.contentEquals(v2)) return false\n v1 is ULongArray  && v2 is ULongArray  -> if (!v1.contentEquals(v2)) return false\n
else  -> if (v1 != v2) return false\n        }

    }
    return true
}

@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepToString")
@kotlin.js.JsName("contentDeepToStringImpl")
internal fun <T> Array<out T>?.contentDeepToStringImpl(): String {
    if (this == null) return "null"
    val length = size.coerceAtMost((Int.MAX_VALUE - 2) / 5) * 5 + 2 // in order not to overflow Int.MAX_VALUE
    return buildString(length) {
        contentDeepToStringInternal(this, mutableListOf())
    }
}

@OptIn(ExperimentalUnsignedTypes::class)
private fun <T> Array<out T>.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<Array<*>>) {
    if (this in processed) {
        result.append("[...]")
        return
    }
    processed.add(this)
    result.append('[')

    for (i in indices) {
        if (i != 0) {
            result.append(', ')
        }
        val element = this[i]
        when (element) {
            null            -> result.append("null")
            is Array<*>     -> element.contentDeepToStringInternal(result, processed)
            is ByteArray    -> result.append(element.contentToString())
            is ShortArray   -> result.append(element.contentToString())
            is IntArray     -> result.append(element.contentToString())
            is LongArray    -> result.append(element.contentToString())
            is FloatArray   -> result.append(element.contentToString())
            is DoubleArray  -> result.append(element.contentToString())
            is CharArray    -> result.append(element.contentToString())
            is BooleanArray -> result.append(element.contentToString())

            is UByteArray   -> result.append(element.contentToString())
            is UShortArray  -> result.append(element.contentToString())
            is UIntArray    -> result.append(element.contentToString())
            is ULongArray   -> result.append(element.contentToString())

            else            -> result.append(element.toString())
        }
    }

    result.append(']')

    processed.removeAt(processed.lastIndex)
}
NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).

Stores the value of the property for the given object in this mutable map.

By default no implicit value is provided for maps and a [NoSuchElementException] is thrown. To create a map with implicit default value use [withDefault] method.

Get the value for the specified key and the implicit default value for this map.

When this map already has an implicit default value provided with a former call to [withDefault], it is being replaced by this call.

Returns a wrapper of this read-only map, having the implicit default value provided with the specified function [defaultValue].

Returns a wrapper of this mutable map, having the implicit default value provided with the specified function [defaultValue].

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

When this map already has an implicit default value provided with a former call to [withDefault], it is being replaced by this call.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.

The default value is used when the original map doesn't contain a value for the key specified *

and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.
override fun containsKey(key: K): Boolean = map.containsKey(key)
override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)
override fun get(key: K): V? = map.get(key)
override val keys: MutableSet<K> get() = map.keys
override val values: MutableList<V> get() = map.values
override fun putAll(from: Map<out K, V>): Boolean = map.putAll(from)
override fun clear() = map.clear()

override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })
}

/*-----------------------------------------------------------------
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 *-------------------------------------------------------------------*/

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.Random

/**
 * Removes a single instance of the specified element from this collection, if it is present.
 * Allows to overcome type-safety restriction of `remove` that requires to pass an element of type `E`.
 * @return `true` if the element has been successfully removed; `false` if it was not present in the collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<out T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)

/**
 * Removes all of this collection's elements that are also contained in the specified collection.
 * Allows to overcome type-safety restriction of `removeAll` that requires to pass a collection of type `Collection<E>`.
 * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<out T>.removeAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).removeAll(elements)

/**
 * Retains only the elements in this collection that are contained in the specified collection.
 * Allows to overcome type-safety restriction of `retainAll` that requires to pass a collection of type `Collection<E>`.
 * @return `true` if any element was removed from the collection, `false` if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<out T>.retainAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).retainAll(elements)

/**
 * Adds the specified element to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(element: T) {
    this.add(element)
}

/**
 * Adds all elements of the given elements collection to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Iterable<T>) {
    this.addAll(elements)
}

/**
 * Adds all elements of the given elements array to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Array<T>) {
    this.addAll(elements)
}

/**
 * Adds all elements of the given elements sequence to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Sequence<T>) {
    this.addAll(elements)
}

/**
 * Removes a single instance of the specified element from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(element: T) {
    this.remove(element)
}

/**
 * Removes all elements contained in the given elements collection from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Collection<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given elements array from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given elements sequence from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) {
    this.removeAll(elements)
}

fun <T> MutableList<in T>.addAll(elements: Collection<T>): Boolean {
    when (elements) {
        is Collection -> return addAll(elements)
        else -> {
            var result: Boolean = false
            for (item in elements)
                result = result or (!this.contains(item))
            return result
        }
    }
}

fun <T> MutableList<in T>.addAll(elements: Array<T>): Boolean {
    return addAll(elements)
}

fun <T> MutableList<in T>.addAll(elements: Sequence<T>): Boolean {
    return addAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Collection<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Array<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Sequence<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Collection<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Array<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Sequence<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Collection<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Array<T>): Boolean {
    return removeAll(elements)
}

fun <T> MutableList<in T>.removeAll(elements: Sequence<T>): Boolean {
    return removeAll(elements)
}
if (add(item)) result = true
return result
    }
}

/**
 * Adds all elements of the given [Collection] to this [MutableCollection].
 */
public fun <T> MutableCollection<in T>.addAll(elements: Collection<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (add(item)) result = true
    }
    return result
}

/**
 * Adds all elements of the given [List] to this [MutableList].
 */
public fun <T> MutableList<T>.addAll(elements: List<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (add(item)) result = true
    }
    return result
}

/**
 * Removes all elements from this [MutableCollection] that are also contained in the given [Collection].
 */
public fun <T> MutableCollection<in T>.removeAll(elements: Collection<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (remove(item)) result = true
    }
    return result
}

/**
 * Removes all elements from this [MutableList] that are also contained in the given [List].
 */
public fun <T> MutableList<T>.removeAll(elements: List<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (remove(item)) result = true
    }
    return result
}

/**
 * Removes the element at the specified [index] from this list.
 * In Kotlin one should use the MutableList.removeAt function instead.
 */
@Deprecated("Use removeAt(index) instead.", ReplaceWith("removeAt(index)"), level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> MutableList<T>.remove(index: Int): T = removeAt(index)

/**
 * Removes the first element from this mutable list and returns that removed element, or throws [NoSuchElementException] if this list is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T> MutableList<T>.removeFirst(): T = if (isEmpty()) throw NoSuchElementException("List is empty.") else removeAt(0)

/**
 * Removes the last element from this mutable list and returns that removed element, or returns `null` if this list is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T> MutableList<T>.removeLast(): T = if (isEmpty()) throw NoSuchElementException("List is empty.") else removeAt(lastIndex)
or returns `null` if this list is empty.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@public fun <T>
MutableList<T>.removeLastOrNull(): T? = if (isEmpty()) null else removeAt(lastIndex)
```

```kotlin
@public fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true) 
```

```kotlin
/**
 * Removes all elements from this [MutableList] that match the given [predicate].
 *
 * @return `true` if any element was removed from this collection, or `false` if all elements were retained and collection was not modified.
 */
@public fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean =
```

```kotlin
private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    if (this !is RandomAccess)
        return (this as MutableList<T>).filterInPlace(predicate, predicateResultToRemove)

    var writeIndex: Int = 0
    for (readIndex in 0..lastIndex) {
        val element = this[readIndex]
        if (predicate(element) == predicateResultToRemove)
            continue

        if (writeIndex != readIndex)
            this[writeIndex] = element

        writeIndex++
    }
    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo writeIndex)
            removeAt(removeIndex)

        return true
    } else {
        return false
    }
}
```

```kotlin
/**
 * Returns a reversed read-only view of the original List.
 * All changes made in the original list will be reflected in the reversed one.
 * @sample samples.collections.ReversedViews.asReversed
 */
@public fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this)

/**
 * Returns a reversed mutable view of the original mutable List.
 * All changes made in the original list will be reflected in the reversed one and vice versa.
 * @sample samples.collections.ReversedViews.asReversedMutable
 */
@kotlin.jvm.JvmName("asReversedMutable")
@public fun <T> MutableList<T>.asReversed(): MutableList<T> = ReversedList(this)
```

```kotlin
/**
 * Builds a [Sequence] lazily yielding values one by one.
 * @see kotlin.sequences.generateSequence
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildFibonacciSequence
 */
@SinceKotlin("1.3")
@public fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(block) 
}
```

```kotlin
@public fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean =
```

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@public fun <T>
MutableList<T>.removeLastOrNull(): T? = if (isEmpty()) null else removeAt(lastIndex)
```
DeprecationLevel.ERROR)|\n@kotlin.internal.InlineOnly\npublic inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence { iterator(builderAction) }\n*\n* Builds an [Iterator] lazily yielding values one by one.\n* @sample samples.collections.Sequences.Building.buildIterator\n* @sample samples.collections.Iterables.Building.iterable\n* @SinceKotlin("1.3")\npublic fun <T> iterator(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Iterator<T> = iterator(builderAction)\n*\n* The scope for yielding values of a [Sequence] or an [Iterator],\n* provides [yield] and [yieldAll] suspension functions.\n* * @see sequence\n* * @see iterator\n* * @sample samples.collections.Sequences.Building.buildSequenceYieldAll\n* @sample samples.collections.Sequences.Building.buildFibonacciSequence\n
}
exceptionalState(): Throwable = when (state) {
        State_Done -> NoSuchElementException()
        State_Failed -> IllegalStateException("Unexpected state of the iterator: $state")
        else -> IllegalStateException("Iterator has failed.")
    }

    override suspend fun yield(value: T) {
        nextValue = value
        state = State_Ready
        return suspendCoroutineUninterceptedOrReturn { c ->
            nextStep = c
        }
    }

    override suspend fun yieldAll(iterator: Iterator<T>) {
        if (!iterator.hasNext()) return
        nextIterator = iterator
        state = State_ManyReady
        return suspendCoroutineUninterceptedOrReturn { c ->
            nextStep = c
        }
    }

    // Completion continuation implementation
    override fun resumeWith(result: Result<Unit>) {
        result.getOrThrow() // just rethrow exception if it is there
        state = State_Done
    }

    override val context: CoroutineContext
        get() = EmptyCoroutineContext
}

package kotlin.collections

internal fun checkWindowSizeStep(size: Int, step: Int) {
    require(size > 0 && step > 0) {
        if (size != step)
            "Both size $size and step $step must be greater than zero."
        else
            "size $size must be greater than zero."
    }
}

    checkWindowSizeStep(size, step)
    return Sequence {
        windowedIterator(iterator(), size, step, partialWindows, reuseBuffer)
    }
}

    if (!iterator.hasNext()) return EmptyIterator
    return iterator<List<T>> {
        val bufferInitialCapacity = size.coerceAtMost(1024)
        val gap = step - size
        if (gap >= 0) {
            var buffer = ArrayList<T>(bufferInitialCapacity)
            var skip = 0
            for (e in iterator) {
                if (skip > 0) {
                    skip -= 1; continue
                }
                buffer.add(e)
                if (buffer.size == size) {
                    yield(buffer)
                    if (reuseBuffer) buffer.clear() else buffer = ArrayList(size)
                    skip = gap
                }
            }
        }
        if (buffer.isNotEmpty()) {
            if (partialWindows || buffer.size == size) yield(buffer)
        }
    }
}

internal class MovingSubList<out E>(private val list: List<E>) : AbstractList<E>(), RandomAccess {
    private var fromIndex: Int = 0
    private var _size: Int = 0

    fun move(fromIndex: Int, toIndex: Int) {
        checkRangeIndexes(fromIndex, toIndex, list.size)
        this.fromIndex = fromIndex
        this._size = toIndex - fromIndex
    }

    override fun get(index: Int): E {
        checkElementIndex(index, _size)
        return list[fromIndex + index]
    }

    override val size: Int get() = _size
}

/**
 * Provides ring buffer implementation.
 * Buffer overflow is not allowed so [add] doesn't overwrite tail but raises an exception.
 * *Anprivate class RingBuffer<T>(private val buffer: Array<T>): AbstractList<T>(), RandomAccess {
 * private var fromIndex: Int = 0
 * private var _size: Int = 0
 * fun move(fromIndex: Int, toIndex: Int) {
 *     this.fromIndex = fromIndex
 *     this._size = toIndex - fromIndex
 *     override fun override fun get(index: Int): T {
 *         checkElementIndex(index, _size)
 * @Suppress("UNCHECKED_CAST")
 *         return buffer[startIndex.forward(index)] as T
 *     }
 *     fun isFull(): Boolean {
 *         return buffer.size >= $buffer.size
 *     }
 * }
 * @Suppress("UNCHECKED_CAST")
 * override fun iterator(): Iterator<T> = object : AbstractIterator<T>() {
 *     @Suppress("UNCHECKED_CAST")
 *     override fun hasNext(): Boolean {
 *         return buffer.size < $buffer.size
 *     }
 *     override fun next(): T {
 *         return buffer[startIndex.forward(1)]
 *     }
 *     override fun computeNext(): Boolean {
 *         if (count == 0) {
 *             done = true
 *         } else {
 *             startIndex = startIndex + 1
 *         }
 *         return done
 *     }
 *     override fun setNext(buffer[index] as T) {
 *         index = startIndex + buffer.size
 *     }
 * }
 * override fun remove(): T = buffer.removeFirst() as T
 * }
 * @Suppress("UNCHECKED_CAST")
 * override fun removeFirst(): T = buffer.removeFirst() as T
 */

result: Array<T?> =
  if (array.size < this.size) array.copyOf(this.size) else array as Array<T?>

  val size = this.size

  var widx = 0
  var idx = startIndex

  while (widx < size && idx < capacity) {
    result[widx] = buffer[idx] as T
    widx++
    idx++
  }

  idx = 0
  while (widx < size) {
    result[widx] = buffer[idx] as T
    widx++
    idx++
  }

  if (result.size > this.size) result[this.size] = null

  return result as Array<T?>

override fun toArray(): Array<Any?> {
  return toArray(arrayOfNulls(size))
}

/**
 * Creates a new ring buffer with the capacity equal to the minimum of [maxCapacity] and 1.5 * [capacity] *
 * The returned ring buffer contains the same elements as this ring buffer.
 * fun expanded(maxCapacity: Int): RingBuffer<T> {
 * This function creates an expanded ring buffer with the specified capacity.
 * If the new capacity is larger than the current size, the buffer will be extended accordingly.
 * If the new capacity is smaller than the current size, the buffer will be truncated to the new capacity.
 * @param maxCapacity The maximum capacity for the new ring buffer.
 * @return A new ring buffer with the specified capacity.
 */

fun expanded(maxCapacity: Int): RingBuffer<T> {
  val newCapacity = (capacity + (capacity shr 1) + 1).coerceAtMost(maxCapacity)
  val newArray = if (startIndex == 0)
    buffer.copyOf(newCapacity)
  else toArray(arrayOfNulls(newCapacity))
  return RingBuffer(newArray, size)
}

/**
 * Add [element] to the buffer or fail with [IllegalStateException] if no free space available in the buffer.
 */

fun add(element: T) {
  if (isFull()) {
    throw IllegalStateException("ring buffer is full")
  }

  buffer[startIndex.forward(size)] = element
  size++
}

/**
 * Removes [n] first elements from the buffer or fails with [IllegalArgumentException] if not enough elements in the buffer to remove.
 */

fun removeFirst(n: Int) {
  require(n >= 0) { "n shouldn't be negative but it is $n" }
  require(n <= size) { "n shouldn't be greater than the buffer size: n = $n, size = $size" }

  if (n > 0) {
    val start = startIndex
    val end = start.forward(n)

    if (start > end) {
      buffer.fill(null, start, capacity)
      buffer.fill(null, 0, end)
    } else {
      buffer.fill(null, start, end)
    }
    startIndex = end
    size -= n
  }
}

private inline fun Int.forward(n: Int): Int = (this + n) % capacity

package kotlin.collections

// UByteArray

@ExperimentalUnsignedTypes
private fun partition(array: UByteArray, left: Int, right: Int): Int {
  var i = left
  var j = right
  val pivot = array[(left + right) / 2]
  while (i <= j) {
    while (array[i] < pivot)
      i++
    while (array[j] > pivot)
      j--
    if (i <= j) {
      val tmp = array[i]
      array[i] = array[j]
      array[j] = tmp
      i++
      j--
    }
  }
  return i
}

@ExperimentalUnsignedTypes
private fun quickSort(array: UByteArray, left: Int, right: Int) {
  val index = partition(array, left, right)
  if (left < index - 1)
    quickSort(array, left, index - 1)
  if (index < right)
    quickSort(array, index, right)
}

// UShortArray

@ExperimentalUnsignedTypes
private fun partition(array: UShortArray, left: Int, right: Int): Int {
  var i = left
  var j = right
  val pivot = array[(left + right) / 2]
  while (i <= j) {
    while (array[i] < pivot)
      i++
    while (array[j] > pivot)
      j--
    if (i <= j) {
      val tmp = array[i]
      array[i] = array[j]
      array[j] = tmp
      i++
      j--
    }
  }
  return i
}

@ExperimentalUnsignedTypes
private fun quickSort(array: UShortArray, left: Int, right: Int) {
  val index = partition(array, left, right)
  if (left < index - 1)
    quickSort(array, left, index - 1)
  if (index < right)
    quickSort(array, index, right)
}

// UIntArray

@ExperimentalUnsignedTypes
private fun partition(array: UIntArray, left: Int, right: Int): Int {
  var i = left
  var j = right
  val pivot = array[(left + right) / 2]
  while (i <= j) {
    while (array[i] < pivot)
      i++
    while (array[j] > pivot)
      j--
    if (i <= j) {
      val tmp = array[i]
      array[i] = array[j]
      array[j] = tmp
      i++
      j--
    }
  }
  return i
}

@ExperimentalUnsignedTypes
private fun quickSort(array: UIntArray, left: Int, right: Int) {
  val index = partition(array, left, right)
  if (left < index - 1)
    quickSort(array, left, index - 1)
  if (index < right)
    quickSort(array, index, right)
}

// ULongArray

@ExperimentalUnsignedTypes
private fun partition(array: ULongArray, left: Int, right: Int): Int {
  var i = left
  var j = right
  val pivot = array[(left + right) / 2]
  while (i <= j) {
    while (array[i] < pivot)
      i++
    while (array[j] > pivot)
      j--
    if (i <= j) {
      val tmp = array[i]
      array[i] = array[j]
      array[j] = tmp
      i++
      j--
    }
  }
  return i
}

@ExperimentalUnsignedTypes
private fun quickSort(array: ULongArray, left: Int, right: Int) {
  val index = partition(array, left, right)
  if (left < index - 1)
    quickSort(array, left, index - 1)
  if (index < right)
    quickSort(array, index, right)
}
var j = right; val pivot = array[(left + right) / 2]; while (i <= j) {
    while (array[i] < pivot) i++;
    while (array[j] > pivot) j--;
    if (i <= j) {
        val tmp = array[i]; array[i] = array[j];
        array[j] = tmp;
        i++; j--;
    }
}
return i;

private fun quickSort(array: ULongArray, left: Int, right: Int) {
    val index = partition(array, left, right);
    if (left < index - 1) quickSort(array, left, index - 1);
    if (index < right) quickSort(array, index, right);
}

/**
 * Sorts the given array using qsort algorithm.
 */
@ExperimentalUnsignedTypes
internal fun sortArray(array: UByteArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: UShortArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: UIntArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: ULongArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)

package kotlin.contracts

@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public interface ContractBuilder {

    /**
     * Describes a situation when a function returns normally, without any exceptions thrown.
     *
     * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.
     *
     * @see sample contracts.returnsContract
     */
    public fun returns(): Returns

    /**
     * Describes a situation when a function returns normally with the specified return [value].
     *
     * The possible values of [value] are limited to `true`, `false` or `null`.
     *
     * @see sample contracts.returnsTrueContract
     * @see sample contracts.returnsFalseContract
     * @see sample contracts.returnsNullContract
     */
    public fun returns(value: Any?): Returns

    /**
     * Describes a situation when a function returns normally with any value that is not `null`.
     *
     * @see sample contracts.returnsNotNullContract
     */
    public fun returnsNotNull(): ReturnsNotNull

    /**
     * Specifies that the function parameter [lambda] is invoked in place.
     *
     * @see sample contracts.callsInPlaceAtMostOnceContract
     * @see sample contracts.callsInPlaceAtLeastOnceContract
     * @see sample contracts.callsInPlaceExactlyOnceContract
     * @see sample contracts.callsInPlaceUnknownContract
     */
    @ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace

    /**
     * Specifies how
many times a function invokes its function parameter in place. See [ContractBuilder.callsInPlace] for the details of the call-in-place function contract.

```kotlin
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public enum class InvocationKind {

    /**
     * A function parameter will be invoked one or more times.
     */
    AT_MOST_ONCE,

    /**
     * A function parameter will be invoked exactly one time.
     */
    AT_LEAST_ONCE,

    /**
     * A function parameter will be invoked exactly one time.
     */
    EXACTLY_ONCE,

    /**
     * A function parameter is called in place, but it's unknown how many times it can be called.
     */
    UNKNOWN
}
```

```kotlin
/**
 * Specifies the contract of a function.
 *
 * The contract description must be at the beginning of a function and have at least one effect.
 * Only the top-level functions can have a contract for now.
 * @param builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.
 */
@ContractsDsl
@ExperimentalContracts
@InlineOnly
@SinceKotlin("1.3")
@Suppress("UNUSED_PARAMETER")
public inline fun contract(builder: ContractBuilder.() -> Unit) { }
```

```kotlin
@SinceKotlin("1.3")
public interface ContinuationInterceptor : CoroutineContext.Element {
    /**
     * The key that defines *the* context interceptor.
     */
    companion object Key : CoroutineContext.Key<ContinuationInterceptor>

    /**
     * Returns continuation that wraps the original continuation, thus intercepting all resumptions.
     * This function is invoked by coroutines framework when needed and the resulting continuations are cached internally per each instance of the original continuation.
     * This function may simply return original continuation if it does not want to intercept this particular continuation.
     */
    public fun <T> interceptContinuation(continuation: Continuation<T>): Continuation<T>

    /**
     * Invoked for the continuation instance returned by [interceptContinuation] when the original continuation completes and will not be used anymore. This function is invoked only if [interceptContinuation] had returned a different continuation instance.
     */
    public fun releaseInterceptedContinuation(continuation: Continuation<*>) {
        /* do nothing by default */
    }

        // getPolymorphicKey specialized for ContinuationInterceptor key
    }
```

---

Open Source Used In DNAC Platform Arica 8280
public override fun minusKey(key: CoroutineContext.Key<*?>): CoroutineContext {
    // minusPolymorphicKey specialized for ContinuationInterceptor key
    @OptIn(ExperimentalStdlibApi::class)
    if (key is AbstractCoroutineContextKey<*?, *>?) {
        return if (key.isSubKey(this.key) && &key.tryCast(this) != null) EmptyCoroutineContext else this
    }
    return if (ContinuationInterceptor === key) EmptyCoroutineContext else this
}

public override fun plus(context: CoroutineContext): CoroutineContext =
    if (context === EmptyCoroutineContext) this else // fast path -- avoid

    make sure interceptor is always last in the context (and thus is fast to get when present)
    val interceptor = removed[ContinuationInterceptor]
    if (interceptor == null) CombinedContext(removed, interceptor) else

    CombinedContext(CombinedContext(left, element), interceptor)
}


containsAll(context: CombinedContext): Boolean {
    var cur = context
    while (true) {
        if (!contains(cur.element)) return false
        val next = cur.left
        if (next is CombinedContext) {
            cur = next
        } else {
            return contains(next as Element)
        }
    }
}

override fun equals(other: Any?): Boolean =
    this === other || other is CombinedContext && other.size() == size() &&
    other.containsAll(this)

override fun hashCode(): Int = left.hashCode() + element.hashCode()

override fun toString(): String =
    "[" + fold("") { acc, element ->
        if (acc.isEmpty()) element.toString() else
            "$acc, $element"
    } + "]"

private fun writeReplace(): Any {
    val n = size()
    val elements =
        arrayOfNulls<CoroutineContext>(n)
    var index = 0
    fold(Unit) { _, element -> elements[index++] = element }
    check(index == n)
    @Suppress("UNCHECKED_CAST")
    return Serialized(elements as Array<CoroutineContext>)
}

private class Serialized(val elements: Array<CoroutineContext>) : Serializable {
    companion object {
        private const val serialVersionUID: Long = 0L
    }

    private fun readResolve(): Any =
        elements.fold(EmptyCoroutineContext, CoroutineContext::plus)
}

@file:kotlin.jvm.JvmName("IntrinsicsKt")
@file:kotlin.jvm.JvmMultifileClass
package kotlin.coroutines.intrinsics

import kotlin.contracts.*
import kotlin.coroutines.*
import kotlin.internal.InlineOnly

/**
 * Obtains the current continuation instance inside suspend functions and either
 * suspends or returns result immediately without suspension.
 * If the [block] returns the special [COROUTINE_SUSPENDED] value, it means that suspend function did suspend the execution
 * and will not return any result immediately. In this case, the [Continuation] provided to the [block] shall be resumed by invoking [Continuation.resumeWith] at some moment in the future when the result becomes available to resume the computation.
 * Otherwise, the return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.
 * It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.
 * As the result type of the [block] is declared as `Any` and cannot be correctly type-checked, its proper return type remains on the conscience of the suspend function's author.
 * Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's thread without going through the [ContinuationInterceptor] that might be present in the coroutine's [CoroutineContext].
 * It is the invoker's responsibility to ensure that a proper invocation context is established.
 * [Continuation.intercepted] can be used to acquire the intercepted continuation.
 * Note that it is not recommended to call either [Continuation.resume] nor [Continuation.resumeWithException] functions synchronously in the same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way to obtain current continuation instance.
 */
@SinceKotlin("1.3")
@InlineOnly
@Suppress("UNUSED_PARAMETER", "RedundantSuspendModifier")
public suspend inline fun <T> suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } throw NotImplemented() 
    suspendCoroutineUninterceptedOrReturn(block)
}

/**
 * This value is used as a return value of [suspendCoroutineUninterceptedOrReturn] `block` argument to state that
 * the execution was suspended and will not return any result immediately.
 * Note: this value should not be used in general code.
 */
@SinceKotlin("1.3")
@PublishedApi // This class is Published API via serialized representation of SafeContinuation, don't rename move\unintenal enum class CoroutineSingletons {
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.experimental

/** Performs a bitwise AND operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.and(other: Byte): Byte = (this.toInt() and other.toInt()).toByte()

/** Performs a bitwise OR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.or(other: Byte): Byte = (this.toInt() or other.toInt()).toByte()

/** Performs a bitwise XOR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.xor(other: Byte): Byte = (this.toInt() xor other.toInt()).toByte()

/** Inverts the bits in this value. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Byte.inv(): Byte = (this.toInt().inv()).toByte()

/** Performs a bitwise AND operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.and(other: Short): Short = (this.toInt() and other.toInt()).toShort()

/** Performs a bitwise OR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.or(other: Short): Short = (this.toInt() or other.toInt()).toShort()

/** Performs a bitwise XOR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Short.xor(other: Short): Short = (this.toInt() xor other.toInt()).toShort()

/** Inverts the bits in this value. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Short.inv(): Short = (this.toInt().inv()).toShort()

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.internal

/**
 * Specifies that the corresponding type should be ignored during type inference.
 */
@Target(AnnotationTarget.TYPE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
@MustBeDocumented
internal annotation class OnlyInputTypes

/**
 * Specifies that this function should not be called directly without inlining.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY_GETTER,
 AnnotationTarget.PROPERTY
)
@Retention(AnnotationRetention.BINARY)
@MustBeDocumented
internal annotation class LowPriorityInOverloadResolution

/**
 * Specifies that a corresponding member has the highest priority in overload resolution.
 */
@Target(AnnotationTarget.FUNCTION,
 AnnotationTarget.PROPERTY
)
@Retention(AnnotationRetention.BINARY)
@MustBeDocumented
internal annotation class HidesMembers

/**
 * The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).
 */
@Target(AnnotationTarget.TYPED_PARAMETER)
@Retention(AnnotationRetention.BINARY)
@MustBeDocumented
internal annotation class OnlyInputTypes

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@Target(AnnotationTarget.FUNCTION,
        AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class DynamicExtension

/**
 * The value of this parameter should be a property reference expression (`this::foo`),
 referencing a `lateinit` property, the backing field of which is accessible at the point where the corresponding
 argument is passed.
 */
@Target(AnnotationTarget.VALUE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.2")
internal annotation class AccessibleLateinitPropertyLiteral

/**
 * Specifies that this declaration is
 * only completely supported since the specified version.
 * The diagnostic message can be specified with [message], or via
 * [errorCode] (takes less space, but might not be immediately clear to the user). The diagnostic severity can be
 * specified with [level]: WARNING/ERROR mean that either a warning or an error is going to be reported,
 * HIDDEN means that the declaration is going to be removed from resolution completely.
 * This specifies version which should be compared with the [version] value, when compiling the usage of the annotated
description. Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported,
 * so the Kotlin compiler before 1.2 is going to treat any [RequireKotlin] as if it requires the language version.
 * Since 1.2, the Kotlin compiler supports
 * [RequireKotlinVersionKind.LANGUAGE_VERSION], [RequireKotlinVersionKind.COMPILER_VERSION] and [RequireKotlinVersionKind.API_VERSION].
 * If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin),
 * Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as before Kotlin 1.2 is going to treat this as a
 * requirement on the language version.
 * This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 * @sample samples.properties
 */
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.2")
internal enum class RequireKotlinVersionKind {
    LANGUAGE_VERSION,
    COMPILER_VERSION,
    API_VERSION,
}

/**
 * Specifies that this declaration is a part of special DSL, used for constructing
 * function's contract.
 */
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.2")
internal annotation class ContractsDsl

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Standard property delegates.
 */
public object Delegates {
    /**
     * Returns a property delegate for a read/write
     * property with a non-null value that is initialized not during
     * object construction time but at a later time.
     *
     * @param initialValue the initial value of the property.
     * @param onChange the callback which is called after the change of the property is made.
     * @return property delegate for a read/write property that calls a specified callback function when changed.
     */
    public inline fun <T> observable(initialValue: T,
            crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):
            ReadWriteProperty<Any?, T> =
            object : ObservableProperty<T>(initialValue) {
                override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)
            }

    /**
     * Returns a property delegate for a read/write property that calls a specified callback function when
     * has already been changed when this callback is invoked.
     */
    public fun <T> onChange(callback: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):
            ReadWriteProperty<Any?, T> =
            object : ObservableProperty<T>(initialValue) {
                override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = callback(property, oldValue, newValue)
            }
    }

    /**
     * Returns a property delegate for a read/write property that calls a specified callback function when
     * has already been changed when this callback is invoked.
     */
    public fun <T> onWriteMaybe(callback: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):
            ReadWriteProperty<Any?, T> =
            object : ObservableProperty<T>(initialValue) {
                override fun afterWrite(property: KProperty<*>, oldValue: T, newValue: T) = callback(property, oldValue, newValue)
            }
    }
}

// Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the
// license/LICENSE.txt file.
// https://github.com/Kotlin/ktor/blob/master/LICENSE

changed, allowing the callback to veto the modification.

* @param initialValue the initial value of the property.

* @param onChange the callback which is called before a change to the property value is attempted.

* The value of the property hasn’t been changed yet, when this callback is invoked.

* If the callback returns "true" the value of the property is being set to the new value, and if the callback returns "false" the new value is discarded and the property remains its old value.

* @sample samples.properties.Delegates.vetoableDelegate

* @sample samples.properties.Delegates.throwVetoableDelegate

* @sample WriteReadProperty<Any?, T> vetoable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Boolean): ObservableProperty<T>(initialValue) {
    override fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = onChange(property, oldValue, newValue)
    //...
/**
 * The callback which is called before a change to the property value is attempted.
 * The value of the property hasn't been changed yet, when this callback is invoked.
 * If the callback returns 'true' the value of the property is being set to the new value,
 * and if the callback returns 'false' the new value is discarded and the property remains its old value.
 */
protected open fun beforeChange(property: KProperty<*>, oldValue: V, newValue: V): Boolean = true
/**
 * The callback which is called after the change of the property is made.
 * The value of the property has already been changed when this callback is invoked.
 */
protected open fun afterChange(property: KProperty<*>, oldValue: V, newValue: V): Unit {}

public override fun getValue(thisRef: Any?, property: KProperty<*>): V {
    return value
}

public override fun setValue(thisRef: Any?, property: KProperty<*>, value: V) {
    val oldValue = this.value
    if (!beforeChange(property, oldValue, value)) {
        return
    }
    this.value = value
    afterChange(property, oldValue, value)
}

package kotlin

import kotlin.reflect.*

/**
 * An extension operator that allows delegating a read-only property of type [V]
 * to a property reference to a property of type [V] or its subtype.
 * An extension operator that allows delegating a read-only property of type [V]
 * to a property reference to a property of type [V] or its subtype.
 * The reference is without a receiver, i.e. it either references a top-level property or a class
 * has the receiver bound to it.
 * The reference is without a receiver, i.e. it either references a top-level property
 * or a class has the receiver bound to it.
 * The reference is without a receiver, i.e. it either references a top-level property
 * or a class has the receiver bound to it.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline operator fun <V> KProperty0<V>.getValue(thisRef: Any?, property: KProperty<*>): V {
    return get(thisRef)
}

/**
 * An extension operator that allows delegating a mutable property of type [V]
 * to a property reference to a property of type [V] or its subtype.
 * An extension operator that allows delegating a mutable property of type [V]
 * to a property reference to a property of type [V] or its subtype.
 * An extension operator that allows delegating a mutable property of type [V]
 * to a property reference to a property of type [V] or its subtype.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline operator fun <V> KMutableProperty0<V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {
    set(thisRef, value)
}
*un * The companion object [Random.Default] is the default instance of [Random].
*un * To get a seeded instance of random generator use [Random] function.
*un * @sample samples.random.Randoms.defaultRandom

*An @SinceKotlin("1.3") public abstract class Random {
    /**
     * Generates an Int whose lower [bitCount] bits are filled with random values and the remaining
     * upper bits are zero.
     */

    public abstract fun nextInt(bitCount: Int): Int

    /**
     * Gets the next random Int from the random number generator.
     */

    public abstract fun nextRandomInt(): Int

    /**
     * Generates an Int random value uniformly distributed between `Int.MIN_VALUE` and `Int.MAX_VALUE`
     * (inclusive).
     */

    public abstract fun nextInt1(): Int

    /**
     * Generates a `Long` random value uniformly distributed between `0` (inclusive) and the specified
     * [until] bound (exclusive).
     */

    public open fun nextLong(until: Long): Long = nextLong(0, until)

    /**
     * Gets the next random `Long` from the random number generator in the specified range.
     */

    public open fun nextRandomLong(): Long

    /**
     * Generates a `Long` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextLong(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Gets the next random `Int` from the random number generator.
     */

    public open fun nextRandomInt(): Int

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Gets the next random `Int` from the random number generator.
     */

    public open fun nextRandomInt(): Int

    /**
     * Generates an Int random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextInt(until: Int): Int =

nextBits(bitCount)

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }

    /**
     * Generates a `Long` random value uniformly distributed between 0 (inclusive) and
     * the specified [until] bound.
     */

    public open fun nextLong(until: Long): Long =

nextBits(bitCount)

    /**
     * Generates an `Int` random value uniformly distributed between the specified [from] (inclusive)
     * and [until] (exclusive) bounds.
     */

    public open fun nextInt(from: Int, until: Int): Int {
        checkRangeBounds(from, until)
        return range from until
    }
the random number generator in the specified range.

* `nextDouble()` Generates a `Double` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.

* `nextInt()` Generates a `Int` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.

* `nextIntUnbounded()`, `nextInt64()`, `nextInt64Unbounded()` Generates a `Long` random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.

* `nextBytes()` Fills a subrange of the specified byte array starting from [from] (inclusive) and ending [toIndex] (exclusive) with random bytes.

* `nextFloat()` Gets the next random `Float` value uniformly distributed between 0 (inclusive) and 1 (exclusive).

* `nextInt()` Gets the next random `Int` value uniformly distributed between 0 (inclusive) and INT_MAX (exclusive).

* `nextLong()` Gets the next random `Long` value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextBoolean()` Gets the next random `Boolean` value.

* `nextDouble()` Gets the next random `Double` value uniformly distributed between 0 (inclusive) and 1 (exclusive).

* `nextBytes()` Fills a subrange of the specified byte array starting from [from] (inclusive) and ending [toIndex] (exclusive) with random bytes.

* `nextBytes()` Fills the specified size [array] with random bytes and returns it.

* `nextInt()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and INT_MAX (exclusive).

* `nextIntUnbounded()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64Unbounded()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and INT_MAX (exclusive).

* `nextIntUnbounded()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64Unbounded()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextBoolean()` Generates a `Boolean` random value.

* `nextDouble()` Generates a `Double` random value uniformly distributed between 0 (inclusive) and 1 (exclusive).

* `nextInt()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and INT_MAX (exclusive).

* `nextIntUnbounded()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64Unbounded()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextBoolean()` Generates a `Boolean` random value.

* `nextDouble()` Generates a `Double` random value uniformly distributed between 0 (inclusive) and 1 (exclusive).

* `nextInt()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and INT_MAX (exclusive).

* `nextIntUnbounded()` Generates a `Int` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextInt64Unbounded()` Generates a `Long` random value uniformly distributed between 0 (inclusive) and LONG_MAX (exclusive).

* `nextBoolean()` Generates a `Boolean` random value.
thread-safe. Do not invoke it from multiple threads without proper synchronization.

```kotlin
*@
@sample
Randoms.seededRandom

*@
@SinceKotlin("1.3")
public fun Random(seed: Long): Random =
XorWowRandom(seed.toInt(), seed.shr(32).toInt())

*@
@SinceKotlin("1.3")
public fun Random.nextInt(range: IntRange): Int = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)
    else -> nextInt(range.first - 1, range.last) + 1
}

*@
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt =
nextInt().toUInt()

*@
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(until: UInt): UInt =
nextUInt(0u, until)

*@
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(from: UInt, until: UInt): UInt {
    checkUIntRangeBounds(from, until)

    val signedFrom = from.toInt() xor Int.MIN_VALUE
    val
    signedUntil = until.toInt() xor Int.MIN_VALUE

    val signedResult = nextInt(signedFrom, signedUntil) xor
    Int.MIN_VALUE
    return signedResult.toUInt()
}
```

IllegalArgumentException if [range] is empty. *@
@SinceKotlin("1.3")
public fun Random.nextInt(range: LongRange): Long = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Long.MAX_VALUE ->
    nextLong(range.first, range.last + 1u)
    else -> nextLong(range.first - 1u, range.last) +
    1u
}

IllegalArgumentException if [range] is empty. *@
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(range: UIntRange): UInt = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < UInt.MAX_VALUE ->
    nextUInt(range.first, range.last + 1u)
    else -> nextUInt(range.first - 1u, range.last) +
    1u
}
```
Open Source Used In DNAC Platform Arica

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.random

/**
 * Random number generator, using Marsaglia's "xorwow" algorithm
 *
 * Cycles after 2^192 - 2^32 repetitions.
 *
 * For more details, see Marsaglia, George (July 2003). "Xorshift RNGs". Journal of Statistical Software. 8, 14, p. 5
 * Available at https://www.jstatsoft.org/v08/i14/paper
 */
internal class XorWowRandom()
internal constructor(
    private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private var addend: Int
) : Random()

    init {
        require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero element." }

        // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64
        repeat(64) { nextInt() }
    }

    override fun nextInt(): Int {
        // Equivalent to the xorwow algorithm
        var t = x
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0 xor t
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        v = t
        addend += 362437
        return t + addend
    }
```

```
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.random

/**
 * Random number generator, using Marsaglia's "xorwow" algorithm
 *
 * Cycles after 2^192 - 2^32 repetitions.
 *
 * For more details, see Marsaglia, George (July 2003). "Xorshift RNGs". Journal of Statistical Software. 8, 14, p. 5
 * Available at https://www.jstatsoft.org/v08/i14/paper
 */
internal class XorWowRandom()
internal constructor(
    private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private var addend: Int
) : Random()

    init {
        require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero element." }

        // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64
        repeat(64) { nextInt() }
    }

    override fun nextInt(): Int {
        // Equivalent to the xorwow algorithm
        var t = x
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0 xor t
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        v = t
        addend += 362437
        return t + addend
    }
```
JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

@file: kotlin.jvm.JvmMultifileClass
@file: kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

/**
 * Represents a range of [Comparable] values.
 */
private open class ComparableRange<T : Comparable<T>>(
    override val start: T,
    override val endInclusive: T
) : ClosedRange<T> {

    override fun equals(other: Any?): Boolean {
        return other is ComparableRange<*> &&
                start == other.start && endInclusive == other.endInclusive
    }

    override fun hashCode(): Int = 31 * start.hashCode() + endInclusive.hashCode()

    override fun toString(): String = "$start..$endInclusive"
}

/**
 * Creates a range from this [Comparable] value to the specified [that] value.
 * @sample samples.ranges.Ranges.rangeFromComparable
 */
public operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)

/**
 * Represents a range of floating point numbers.
 * Extends [ClosedRange] interface providing custom operation
 * [lessThanOrEquals] for comparing values of range domain type.
 * @sample samples.ranges.Ranges.rangeFromDouble
 */
@SinceKotlin("1.1")
public interface ClosedFloatingPointRange<T : Comparable<T>> : ClosedRange<T> {
    override fun contains(value: T): Boolean = lessThanOrEquals(start, value) &&
            lessThanOrEquals(value, endInclusive)
    override fun isEmpty(): Boolean = !lessThanOrEquals(start, endInclusive)

    /**
     * Compares two values of range domain type and returns true if first
     * is less than or equal to second.
     */
    fun lessThanOrEquals(a: T, b: T): Boolean
}

/**
 * A closed range of values of type `Double`.
 * Numbers are compared with the ends of this range according to
 * IEEE-754.
 */
private class ClosedDoubleRange(
    start: Double,
    endInclusive: Double
) : ClosedFloatingPointRange<Double> {
    private val _start = start
    private val _endInclusive = endInclusive

    override val start: Double get() = _start
    override val endInclusive: Double get() = _endInclusive

    override fun lessThanOrEquals(a: Double, b: Double): Boolean = a <= b

    override fun contains(value: Double): Boolean = value >= _start &&
            value <= _endInclusive
    override fun isEmpty(): Boolean = !_start <= _endInclusive

    override fun equals(other: Any?): Boolean {
        return other is ClosedDoubleRange &&
                _start == other._start && _endInclusive == other._endInclusive
    }

    override fun hashCode(): Int = 31 * _start.hashCode() + _endInclusive.hashCode()

    override fun toString(): String = "$_start..$_endInclusive"
}

/**
 * Creates a range from this [Double] value to the specified [that] value.
 * Numbers are compared with the ends of this range according to
 * IEEE-754.
 */
@SinceKotlin("1.1")
public operator fun Double.rangeTo(that: Double): ClosedFloatingPointRange<Double> = ClosedDoubleRange(this, that)

/**
 * Returns `true` if this iterable range contains the specified [element].
 * Always returns `false` if the [element] is `null`.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline operator fun <T, R> R.contains(element: T?):
covariant projection of the type of class `Number`.

@JvmStatic
public fun covariant(type: KType): KTypeProjection =

KTypeProjection(KVariance.OUT, type)

/**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect

/**
 * Represents variance applied to a type parameter on the declaration site (*declaration-site variance*),
 * or to a type in a projection (*use-site variance*).
 */

@SinceKotlin("1.1")
enum class KVariance
{
    /**
     * The affected type parameter or type is *invariant*, which means it has no variance applied to it.
     */
    INVARIANT,

    /**
     * The affected type parameter or type is *contravariant*. Denoted by the `in` modifier in the source code.
     */
    IN,

    /**
     * The affected type parameter or type is *covariant*. Denoted by the `out` modifier in the source code.
     */
    OUT,
}

Returns a runtime representation of the given reified type [T] as an instance of [KType].

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <reified T> typeOf(): KType =

throw UnsupportedOperationException("This function is implemented as an intrinsic on all supported platforms.")

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

/**
 * Appends all arguments to the given Appendable.
 */
public fun <T : Appendable> T.append(vararg value: CharSequence?): T {
    for (item in value)
        append(item)
    return this
}

/** Appends a line feed character (`\n`) to this Appendable and returns this instance.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Appendable.appendLine(): Appendable = append('\n')

/** Appends value to the given Appendable and a line feed character (`\n`) after it.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Appendable.appendLine(value: CharSequence?): Appendable = append(value).appendLine()
to every line of the original string.

String.prependIndent(indent: String = "\"") { String =\n    lineSequence() .map { it when { it isBlank() -> \n                         \n                         when { it .length < indent .length .then - indent \n                         \n                         else .-> indent + it \n                         \n                         } .joinToString(\"\n\") } } .unprivate fun

String.insertIndentWidth(): Int = indexOfFirst { it isWhitespace() } .let { if (it == -1) length else it } .unprivate fun

getIndentFunction(indent: String) = when { it.isBlank() -> \n    when { it .length < indent .length .then - indent \n    \n    \n    } \n    \n    else -> indent + line \n    \n    } .unprivate inline list< String > \n    .reindent\n
private fun String.indentWidth(): Int = indexOfFirst { !it .isWhitespace() } .let { if (it == -1) length else it } .unprivate fun

getIndentFunction(indent: String) = when { it.isBlank() -> \n    when { it .length < indent .length .then - indent \n    \n    \n    } \n    \n    else -> indent + line \n    \n    } .unprivate inline list< String > \n    .reindent\n
private fun getIndentFunction(indent: String) = when { it.isBlank() -> \n    when { it .length < indent .length .then - indent \n    \n    \n    } \n    \n    else -> indent + line \n    \n    } .unprivate inline list< String > \n    .reindent\n
private fun String.indentCutFunction: String -> String? = String { it .lastIndex == lastIndex } .return mapIndexedNotNull { index, value -> \n    if (index == 0 || index == lastIndex) \&& value .isBlank() } .unnull else \n    indentCutFunction(value) \?.let(\n    indentAddFunction) ?: \n    \n    } .unprivate fun

joinToString(StringBuilder(resultSizeEstimate), "\n") .unpackage kotlin.text

package kotlin.text

/**
 * Defines names for Unicode symbols used in proper Typography.
 */
public object Typography {
    /** The character " quoting mark */
    public const val quote: Char = '\u0022'
    /** The character $ dollar sign */
    public const val dollar: Char = '\u0024'
    /** The character & ampersand */
    public const val amp: Char = '\u0026'
    /** The character < less-than sign */
    public const val less: Char = '\u003C'
    /** The character > greater-than sign */
    public const val greater: Char = '\u003E'
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The character × times sign */
    public const val times: Char = '\u00D7'
    /** The character ¢ cent */
    public const val cent: Char = '\u00A2'
    /** The character £ pound sign */
    public const val pound: Char = '\u00A3'
    /** The character § section sign */
    public const val section: Char = '\u00A7'
    /** The character © copyright sign */
    public const val copyright: Char = '\u00A9'
    /** The character « left guillemete */
    public const val leftGuillemete: Char = '\u00AB'
    /** The character » right guillemete */
    public const val rightGuillemete: Char = '\u00BB'
    /** The character ® registered sign */
    public const val registered: Char = '\u00AE'
    /** The character ° degree */
    public const val degree: Char = '\u00B0'
    /** The character ± plus or minus sign */
    public const val plusMinus: Char = '\u00B1'
    /** The character ¶ paragraph sign */
    public const val paragraph: Char = '\u00B6'
    /** The character · middle dot */
    public const val middleDot: Char = '\u00B7'
    /** The character ½ half */
    public const val half: Char = '\u00BD'
    /** The character – minus sign */
    public const val minus: Char = '\u00BF'
    /** The character † dagger sign */
    public const val dagger: Char = '\u2020'
    /** The character ‡ double dagger */
    public const val doubleDagger: Char = '\u2021'
    /** The character • bullet */
    public const val bullet: Char = '\u2022'
    /** The character ‼ ellipsis */
    public const val ellipsis: Char = '\u2026'
    /** The character ′ prime sign */
    public const val prime: Char = '\u2032'
    /** The character ″ double prime */
    public const val doublePrime: Char = '\u2033'
    /** The character € euro sign */
    public const val euro: Char = '\u20AC'
    /** The character ≈ almost equal sign */
    public const val almostEqual: Char = '\u2248'
    /** The character ≠ not equal sign */
    public const val notEqual: Char = '\u2260'
    /** The character ≤ less than or equal to sign */
    public const val lessThanOrEqual: Char = '\u2264'
    /** The character ≥ greater than or equal to sign */
    public const val greaterThanOrEqual: Char = '\u2265'
}
An element of the collection at the particular index can be `null`. If the corresponding group in the regular expression is optional and there was no match captured by that group.

```java
public interface MatchGroupCollection : Collection<MatchGroup?> {
    /** Returns a group with the specified [index].
     * Groups are indexed from 1 to the count of groups in the regular expression. A group with the index 0
     * corresponds to the entire match.
     */
    public operator fun get(index: Int): MatchGroup?
}

/**
 * Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.
 */
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
    /**
     * Returns a named group with the specified [name].
     * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.
     * @throws IllegalArgumentException if there is no group with the specified [name] defined in the regex pattern.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public operator fun get(name: String): MatchGroup?
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /** The range of indices in the original string where match was captured. */
    public val range: IntRange
    /** The substring from the input string captured by this match. */
    public val value: String
    /**
     * A collection of groups matched by the regular expression.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * @return A list of matched indexed group values.
     * @throws UnsupportedOperationException if getting indexed groups isn't supported on the current platform.
     */
    public val groups: MatchGroupCollection
    /**
     * A list of matched indexed group values.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * @return A list of matched indexed group values.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public val groupValues: List<String>

    /**
     * An instance of [MatchResult.Destructured] wrapper providing components for destructuring assignment of group values.
     * @return An instance of [MatchResult.Destructured] wrapper providing components for destructuring assignment of group values.
     */
    public val destructured: Destructured get() = Destructured(this)
}

public class Destructured internal constructor(public val match: MatchResult) {
    @kotlin.internal.InlineOnly
    public operator inline fun component1(): String = match.groupValues[1]
    @kotlin.internal.InlineOnly
    public operator inline fun component2(): String = match.groupValues[2]
    @kotlin.internal.InlineOnly
    public operator inline fun component3(): String = match.groupValues[3]
    @kotlin.internal.InlineOnly
    public operator inline fun component4(): String = match.groupValues[4]
    @kotlin.internal.InlineOnly
    public operator inline fun component5(): String = match.groupValues[5]
    @kotlin.internal.InlineOnly
    public operator inline fun component6(): String = match.groupValues[6]
    @kotlin.internal.InlineOnly
    public operator inline fun component7(): String = match.groupValues[7]
    @kotlin.internal.InlineOnly
    public operator inline fun component8(): String = match.groupValues[8]
    @kotlin.internal.InlineOnly
    public operator inline fun component9(): String = match.groupValues[9]
    @kotlin.internal.InlineOnly
    public operator inline fun component10(): String = match.groupValues[10]
}

/**
 * Provides components for destructuring assignment of group values.
 * If the group in the regular expression is optional and there were no match captured by that group,
 * corresponding component value is an empty string.
 */
@sample samples.text.Regexps.matchDestructuringToGroupValues
public class Destructured internal constructor(public val match: MatchResult) {
    @kotlin.internal.InlineOnly
    public operator inline fun component1(): String = match.groupValues[1]
    @kotlin.internal.InlineOnly
    public operator inline fun component2(): String = match.groupValues[2]
    @kotlin.internal.InlineOnly
    public operator inline fun component3(): String = match.groupValues[3]
    @kotlin.internal.InlineOnly
    public operator inline fun component4(): String = match.groupValues[4]
    @kotlin.internal.InlineOnly
    public operator inline fun component5(): String = match.groupValues[5]
    @kotlin.internal.InlineOnly
    public operator inline fun component6(): String = match.groupValues[6]
    @kotlin.internal.InlineOnly
    public operator inline fun component7(): String = match.groupValues[7]
    @kotlin.internal.InlineOnly
    public operator inline fun component8(): String = match.groupValues[8]
    @kotlin.internal.InlineOnly
    public operator inline fun component9(): String = match.groupValues[9]
    @kotlin.internal.InlineOnly
    public operator inline fun component10(): String = match.groupValues[10]

    /**
     * Returns destructured group values as a list of strings.
     * First value in the returned list corresponds to the value of the first group, and so on.
     * @return A list of strings.
     */
    public fun toList(): List<String> = match.groupValues.asList()
}
```

---

Open Source Used In DNAC Platform Arica 8297

---

* Copyright 2010-2020
JetBrains s.r.o. and Kotlin Programming Language contributors.* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.* \nprivate inline val storageUnit

/**
 * Represents the amount of time one instant of time is away from
 * another instant.\n * A negative duration is possible in a situation when the second instant is earlier than the first one.\n * An infinite duration value [Duration.INFINITE] can be used to represent infinite timeouts.\n */
@SinceKotlin("1.3")
@ExperimentalTime
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_O
F_INLINE_CLASS")
public inline class Duration internal constructor(internal val value: Double) :

// backend fails on init block, wait for KT-28055\n//    init {
//        require(_value.isNaN().not())
//    }

    companion object {
        /** The duration equal to exactly 0 seconds. */
        public val ZERO: Duration = Duration(0.0)

        /** The duration whose value is positive infinity. It is
 * useful for representing timeouts that should never expire. */
        public val INFINITE: Duration =
            Duration(Double.POSITIVE_INFINITY)

        /** Converts the given time duration [value] expressed in the
 * specified [sourceUnit] into the specified [targetUnit]. */
        public fun convert(value: Double, sourceUnit:
            DurationUnit): Duration =
            convertDurationUnit(value, sourceUnit, targetUnit)

        public fun plus(other: Duration): Duration = Duration(value + other.value)

        public fun minus(other: Duration): Duration = Duration(value - other.value)

        public fun times(scale: Int): Duration =
            Duration(value * scale)

        public fun times(scale: Double): Duration =
            Duration(value * scale)

        public fun div(scale: Int): Duration =
            Duration(value / scale)

        public fun div(scale: Double): Duration =
            Duration(value / scale)

        public fun div(other: Duration): Double = this.value / other.value

        public fun isNegative(): Boolean = value < 0

        public fun isPositive(): Boolean = value > 0

        public fun isInfinite(): Boolean = value.isInfinite()

        public fun isFinite(): Boolean = value.isFinite()

        public val absoluteValue: Duration get() = if (isNegative()) -this else this

        override fun compareTo(other: Duration): Int = this.value.compareTo(other.value)

    // splitting to components

    /**
     * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action]
     * with these components.\n     * The result of [action] is returned as the result of this function.\n     * \n     * - `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than
     *   1_000_000_000;\n     * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than
     *   60;\n     * - `minutes` represents the whole number of minutes in this duration, and its absolute value is less than
     *   60;\n     * - `hours` represents the whole number of hours in this duration, and its absolute value is less than
     *   24;\n     * - `days` represents the whole number of days in this duration.\n     * If the value doesn’t fit in [Int] range, i.e. it’s greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],\n     * it is coerced into that range.\n     */
    */

    internal fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {
        contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
action(inDays.toInt(), hoursComponent, minutesComponent, secondsComponent, nanosecondsComponent)

    /**
     * Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
     * The result of [action] is returned as the result of this function.
     * * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
     * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
     * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
     * `hours` represents the whole number of hours in this duration.
     * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
     * it is coerced into that range.
     */
    public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {
        contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
        return action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)
    }

    /**
     * Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these components.
     * The result of [action] is returned as the result of this function.
     * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
     * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
     * `minutes` represents the whole number of minutes in this duration.
     * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
     * it is coerced into that range.
     */
    public inline fun <T> toComponents(action: (minutes: Int, seconds: Int, nanoseconds: Int) -> T): T {
        contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
        return action(inMinutes.toInt(), secondsComponent, nanosecondsComponent)
    }

    /**
     * Splits this duration into seconds, and nanoseconds and executes the given [action] with these components.
     * The result of [action] is returned as the result of this function.
     * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
     * `seconds` represents the whole number of seconds in this duration.
     * If the value doesn't fit in [Long] range, i.e. it's greater than [Long.MAX_VALUE] or less than [Long.MIN_VALUE],
     * it is coerced into that range.
     */
    public inline fun <T> toComponents(action: (seconds: Long, nanoseconds: Int) -> T): T {
        contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) }
        return action(inSeconds.toLong(), nanosecondsComponent)
    }

    // conversion to units

    /** Returns the value of this duration expressed as a [Double] number of the specified [unit]. */
    public fun toDouble(unit: DurationUnit): Double = convertDurationUnit(value, storageUnit, unit)

    /** Returns the value of this duration expressed as a [Long] number of the specified [unit].
     * If the value doesn't fit in [Long] range, it is coerced into that range, see the conversion [Double.toLong] for details.
     */
    public fun toLong(unit: DurationUnit): Long = toDouble(unit).toLong()

    /** Returns the value of this duration expressed as an [Int] number of the specified [unit].
     * If the value doesn't fit in the range of [Int] type, it is coerced into that range, see the conversion [Double.toInt] for details.
     */
    public fun toInt(unit: DurationUnit): Int = toDouble(unit).toInt()
Double get() = toDouble(DurationUnit.NANOSECONDS)\n\n // shortcuts\n\n /**\n * Returns the value of
this duration expressed as a [Long] number of nanoseconds.\n *\n * If the value doesn't fit in the range of
[Long] type, it is coerced into that range, see the conversion [Double.toLong] for details.\n *\n * The range of
durations that can be expressed as a `Long` number of nanoseconds is approximately \u00b1292 years.\n */\n
public fun toLongNanoseconds(): Long = toLong(DurationUnit.NANOSECONDS)\n\n /**\n * Returns the
value of this duration expressed as a [Long] number of milliseconds.\n *\n * The value is coerced to the range
of [Long] type, if it doesn't fit in that range, see the conversion [Double.toLong] for details.\n *\n * The range
of durations that can be expressed as a `Long` number of milliseconds is approximately \u00b1292 million years.\n
*/\n public fun toLongMilliseconds(): Long = toLong(DurationUnit.MILLISECONDS)\n\n /**\n * Returns a
string representation of this duration value expressed in the unit which yields the most compact and readable number
value.\n *\n * Special cases:\n * - zero duration is formatted as `\"0s\"`\n * - the infinite duration is
formatted as `\"Infinity\"` without unit\n * - very small durations (less than 1e-15 s) are expressed in seconds and
formatted in scientific notation\n * - very big durations (more than 1e+7 days) are expressed in days and
formatted in scientific notation\n *\n * @return the value of duration in the automatically determined unit
followed by that unit abbreviated name: `d`, `h`, `m`, `s`, `ms`, `us`, or `ns`.\n *\n * @sample
samples.time.Durations.toStringDefault\n */\n override fun toString(): String = when {\n
isInfinite() ->
value.toString()\n
value == 0.0 -> \"0s\"\n
else -> {\n
val absNs = absoluteValue.inNanoseconds\n
var scientific = false\n
var maxDecimals = 0\n
val unit = when {\n
absNs < 1e-6 ->
DurationUnit.SECONDS.also { scientific = true }\n
absNs < 1 -> DurationUnit.NANOSECONDS.also {
maxDecimals = 7 }\n
absNs < 1e3 -> DurationUnit.NANOSECONDS\n
absNs < 1e6 ->
DurationUnit.MICROSECONDS\n
absNs < 1e9 -> DurationUnit.MILLISECONDS\n
absNs <
1000e9 -> DurationUnit.SECONDS\n
absNs < 60_000e9 -> DurationUnit.MINUTES\n
absNs <
3600_000e9 -> DurationUnit.HOURS\n
absNs < 86400e9 * 1e7 -> DurationUnit.DAYS\n
else ->
DurationUnit.DAYS.also { scientific = true }\n
}\n
val value = toDouble(unit)\n
when {\n
scientific -> formatScientific(value)\n
maxDecimals > 0 -> formatUpToDecimals(value,
maxDecimals)\n
else -> formatToExactDecimals(value, precision(abs(value)))\n
}+
unit.shortName()\n
}\n }\n\n private fun precision(value: Double): Int = when {\n
value < 1 -> 3\n
value < 10 -> 2\n
value < 100 -> 1\n
else -> 0\n }\n\n /**\n * Returns a string representation of this
duration value expressed in the given [unit]\n * and formatted with the specified [decimals] number of digits after
decimal point.\n *\n * Special cases:\n * - the infinite duration is formatted as `\"Infinity\"` without unit\n
*\n * @return the value of duration in the specified [unit] followed by that unit abbreviated name: `d`, `h`, `m`,
`s`, `ms`, `us`, or `ns`.\n *\n * @throws IllegalArgumentException if [decimals] is less than zero.\n *\n *
@sample samples.time.Durations.toStringDecimals\n */\n public fun toString(unit: DurationUnit, decimals: Int
= 0): String {\n
require(decimals >= 0) { \"decimals must be not negative, but was $decimals\" }\n
if
(isInfinite()) return value.toString()\n
val number = toDouble(unit)\n
return when {\n
abs(number) <
1e14 -> formatToExactDecimals(number, decimals.coerceAtMost(12))\n
else -> formatScientific(number)\n
} + unit.shortName()\n }\n\n\n /**\n * Returns an ISO-8601 based string representation of this duration.\n
*\n * The returned value is presented in the format `PThHmMs.fS`, where `h`, `m`, `s` are the integer components
of this duration (see [toComponents])\n * and `f` is a fractional part of second. Depending on the roundness of the
value the fractional part can be formatted with either\n * 0, 3, 6, or 9 decimal digits.\n *\n * If the hours
component absolute value of this duration is greater than [Int.MAX_VALUE], it is replaced with
[Int.MAX_VALUE],\n * so the infinite duration is formatted as `\"PT2147483647H\".\n *\n * Negative
durations are indicated with the sign `-` in the beginning of the returned string, for example, `\"-PT5M30S\"`.\n
*\n * @sample samples.time.Durations.toIsoString\n */\n @OptIn(ExperimentalStdlibApi::class)\n public
fun toIsoString(): String = buildString {\n
if (isNegative()) append('-')\n
append(\"PT\")\n
absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->\n
val hasHours = hours != 0\n
val hasSeconds = seconds != 0 || nanoseconds != 0\n
val hasMinutes = minutes != 0 || (hasSeconds &&
hasHours)\n
if (hasHours) {\n
append(hours).append('H')\n
}\n
if (hasMinutes) {\n

Open Source Used In DNAC Platform Arica 8300


append(minutes).append('M')
            }
            if (hasSeconds || (!hasHours && !hasMinutes)) {
                append(seconds)
                if (nanoseconds != 0) {
                    appendRange(nss, 0, 3)
                    val nss =
                    nanoseconds.toString().padStart(9, '0')
                    when {
                        nanoseconds % 1_000_000 == 0 ->
                            appendRange(nss, 0, 3)
                        nanoseconds % 1_000 == 0 -> appendRange(nss, 0, 6)
                        else -> append(nss)
                    }
                }
                append('S')
            }
        }
    }

}

// constructing
// extension functions

/** Returns a [Duration] equal to this [Int] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Int.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Long] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Long.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Double] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Double.toDuration(unit: DurationUnit): Duration =
    Duration(convertDurationUnit(this, unit, storageUnit))

// constructing from number of units
// extension properties

/** Returns a [Duration] equal to this [Int] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Long] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Double] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Int] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Long] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Double] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Int] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Long] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Double] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Int] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Long] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Double] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Int] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Long] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Double] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Int] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.hours get() =
    toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Long] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.hours get() =
    toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Double] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.hours get() =

toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Int] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.days get() =
toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Long] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.days get() =
toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Double] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.days get() =
toDuration(DurationUnit.DAYS)

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
@kotlin.internal.InlineOnly
public inline operator fun Int.times(duration: Duration): Duration = duration * this

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
@kotlin.internal.InlineOnly
public inline operator fun Double.times(duration: Duration): Duration = duration * this

internal expect fun formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double, decimals: Int): String
internal expect fun formatScientific(value: Double): String

@file:kotlin.jvm.JvmMultifileClass()
contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

** @package kotlin.time

* The duration of time interval [elapsed][TimeMark.elapsedNow] from that point. *

@SinceKotlin("1.3")
@ExperimentalTime
public interface TimeSource {

* Marks a point in time on this time source. *

* The returned [TimeMark] instance encapsulates the captured time point and allows querying

* This time source returns its readings from a source of monotonic time when it is available in a target platform. *

and resorts to a non-monotonic time source otherwise. *

* public fun markNow(): TimeMark */

* The most precise time source available in the platform. *

Remains bound to the time source from which it was taken from and allows querying for the duration of time elapsed from that point (see the function [elapsedNow]). *

* Represents a time point notched on a particular [TimeSource]. *

Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* Returns a time mark on the same time source that is ahead of this time mark by the specified [duration]. *

* The returned time mark is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative. *

* Returns a time mark on the same time source that is behind this time mark by the specified [duration]. *

* * The returned time mark is more _early_ when the [duration] is positive, and more _late_ when the [duration] is negative. *

* Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns true if this time mark has passed according to the time source from which this mark was taken. *

* Note that the value returned by this function can change on subsequent invocations. *

* * Returns false if this time mark has not passed according to the time source from which this mark was taken. *
ClockMark = TimeMark\n";`*/n * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*\n
package kotlin.time\n@SinceKotlin("1.3")\n@ExperimentalTime\n


import object MonotonicTimeSource : TimeSource\n/**
 * An abstract class used to implement time sources that return their readings as [Long] values in the specified [unit].
 */

internal

@SinceKotlin("1.3")
@ExperimentalTime
internal

abstract class AbstractLongTimeSource(protected val unit: DurationUnit) : TimeSource {
    /**
     * This protected method should be overridden to return the current reading of the time source expressed as a [Long] number in the unit specified by the [unit] property.
     */
    protected abstract fun read(): Long

    private class LongTimeMark(private val startedAt: Long, private val timeSource: AbstractLongTimeSource, private val offset: Duration) : TimeMark() {
        override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
        override fun plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)
    }

    override fun markNow(): TimeMark = LongTimeMark(read(), this, Duration.ZERO)
}

/**
 * An abstract class used to implement time sources that return their readings as [Double] values in the specified [unit].
 */

public abstract class AbstractDoubleTimeSource(protected val unit: DurationUnit) : TimeSource {
    /**
     * This protected method should be overridden to return the current reading of the time source expressed as a [Double] number in the unit specified by the [unit] property.
     */
    protected abstract fun read(): Double

    private class DoubleTimeMark(private val startedAt: Double, private val timeSource: AbstractDoubleTimeSource, private val offset: Duration) : TimeMark() {
        override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
        override fun plus(duration: Duration): TimeMark = DoubleTimeMark(startedAt, timeSource, offset + duration)
    }

    override fun markNow(): TimeMark = DoubleTimeMark(read(), this, Duration.ZERO)
}

/**
 * A time source that has programmatically updatable readings. It is useful as a predictable source of time in tests.
 */

@SinceKotlin("1.3")
@ExperimentalTime
public class TestTimeSource : AbstractLongTimeSource(unit = DurationUnit.NANOSECONDS) {
    private var reading: Long = 0L

    override fun read(): Long = reading

    /**
     * Advances the current reading value of this time source by the specified [duration].
     *
     * [duration] value is rounded down towards zero when converting it to a [Long] number.
     * For example, if the duration being added is `0.6.nanoseconds`, the reading doesn't advance because the duration value is rounded down to zero nanoseconds.
     * Therefore, the reading value is stored as a [Long] number.
     * If the reading value overflows as the result of this operation, an [IllegalStateException] is thrown.
     */
    public operator fun plusAssign(duration: Duration) {
        val delta = duration.toDouble(unit)
        val longDelta = delta.toLong()

        reading += longDelta
        if (reading xor longDelta >= 0 && reading xor newReading < 0) throw IllegalStateException("TestTimeSource will overflow if its reading $reading is advanced by $duration.")

    }
}

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("use TimeSource.Monotonic instead.", DeprecationLevel.ERROR)
public

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeSource.Monotonic instead.", ReplaceWith("TimeSource.Monotonic\", time.time.TimeSource).\n

public time alias MonoClock =


AbstractLongTimeSource instead.

ReplaceWith("AbstractLongTimeSource\n\"), DeprecationLevel.ERROR\n
public typealias AbstractClock =

AbstractLongTimeSource\n
@SinceKotlin("1.3\n\")\n
@ExperimentalTime\n
@Deprecated("Use

AbstractDoubleTimeSource instead.\n\"), ReplaceWith("\n\AbstractDoubleTimeSource\n\")\n
@SinceKotlin("1.3\n\")\n
@ExperimentalTime\n
@Deprecated("Use

AbstractDoubleTimeSource instead.\n\"), ReplaceWith("\n\AbstractDoubleTimeSource\n\")\n
@SinceKotlin("1.3\n\")\n
@ExperimentalTime\n
@Deprecated("Use

AbstractDoubleTimeSource instead.\n\"), ReplaceWith("\n\AbstractDoubleTimeSource\n\")\n
@SinceKotlin("1.3\n\")\n
@ExperimentalTime\n
@Deprecated("Use

AbstractDoubleTimeSource instead.\n\"), ReplaceWith("\n\AbstractDoubleTimeSource\n\")

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use

AbstractDoubleTimeSource instead.\n\"), ReplaceWith("\n\AbstractDoubleTimeSource\n\")

@SinceKotlin("1.3")
@ExperimentalTime
@DeprecationLevel.ERROR

public typealias AbstractLongClock =

AbstractLongTimeSource

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use

AbstractDoubleClock instead.
\", ReplaceWith("\n\AbstractDoubleClock\n\")

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use

TestClock instead.
\", ReplaceWith("\n\TestClock\n\")

@SinceKotlin("1.3")
@ExperimentalTime
@DeprecationLevel.ERROR

public typealias TestClock =

TestTimeSource

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import kotlin.contracts.*

/**
 * Executes the given
 * function \[block\] and returns the duration of elapsed time interval.\n *\n * The elapsed time is measured with [TimeSource.Monotonic].\n */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun

measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }
    return TimeSource.Monotonic.measureTime(block)\n}

/**
 * Executes the given function \[block\] and returns the duration of elapsed time interval.
 *\n * The elapsed time is measured with the specified \[TimeSource\] instance.
 */
@SinceKotlin("1.3")\n
@ExperimentalTime
public inline fun TimeSource.measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }

    val mark = markNow()\n    block()\n    return mark.elapsedNow()\n}

/**
 * Represents a result of executing an action along with the duration of elapsed time interval.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given function \[block\] and returns an instance of [TimedValue] class, containing
 * both
 * \n * the result of the function execution and the duration of elapsed time interval.\n *\n * The elapsed time is measured with [TimeSource.Monotonic].\n */
@SinceKotlin("1.3")\n
@ExperimentalTime
public inline fun <T> measureTimedValue(block: () -> T): TimedValue<T> {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }

    val mark = markNow()\n    val result = block()\n    return TimedValue(result, mark.elapsedNow())\n}

/**
 * Provides a function to measure an instance of [TimedValue] class, containing
 * both
 * \n * the result of function execution and the duration of elapsed time interval.
 *\n * The elapsed time is measured with the specified \[TimeSource\] instance.
 */
@SinceKotlin("1.3")\n
@ExperimentalTime
public inline fun TimeSource.measureTimedValue(block: () -> TimedValue)\n
/**
 * Defines deep recursive function
 * that keeps its stack on the heap,\n * which allows very deep recursive computations that do not use the actual call
 * stack.\n * To initiate a call to this deep recursive function use its [invoke] function.\n *\n * Note that all argument types of a recursive function should be known at the time
 * of its definition.
 *\n * Example:
 *\n * ```kotlin
 * class Tree(val left: Tree? = null, val right: Tree? = null)
 * val deepTree = generateSequence(Tree()) { Tree(it) }.take(100_000).last()
 * ```
 *\n * ```kotlin
 * fun depth(t: Tree?): Int =
 *     if (t == null) 0 else max(depth(t.left), depth(t.right)) + 1
 * ```
 *\n * If this `depth` function is called for a `deepTree` it produces [StackOverflowError] because of deep recursion.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public typealias DeepRecursiveFunction =

DeepRecursiveScope.callRecursive

/**
 * Deep recursive scope that allows for deep recursive computations.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public interface DeepRecursiveScope

/**
 * Represents a result of deep recursive function.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class DeepRecursiveResult<T>

/**
 * Base class for deep recursive functions.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public abstract class DeepRecursiveFunctionBase<T>
However, the `depth` function can be rewritten using `DeepRecursiveFunction` in the following way, and then it successfully computes `depth(deepTree)` expression:

```kotlin
val depth = DeepRecursiveFunction { t ->
    if (t == null) 0 else max(callRecursive(t.left), callRecursive(t.right)) + 1
}
println(depth(deepTree)) // Ok
```

Deep recursive functions can also mutually call each other using a heap for the stack via `DeepRecursiveFunction.callRecursive` extension. For example, the following pair of mutually recursive functions computes the number of tree nodes at even depth in the tree:

```kotlin
val even:
    DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->
        if (t == null) 0 else odd.callRecursive(t.left) + odd.callRecursive(t.right) + 1
    }
    val odd: DeepRecursiveFunction<Tree?, Int> = DeepRecursiveFunction { t ->
        if (t == null) 0 else even.callRecursive(t.left) + even.callRecursive(t.right)
    }
```

@SinceKotlin("1.4")
@ExperimentalStdlibApi
public class DeepRecursiveFunction<T extends Any?, R extends Any>(
    internal val block: suspend DeepRecursiveScope<T, R>.(T) -> R
) {

    /**
     * Initiates a call to this deep recursive function, forming a root of the call tree.
     * This operator should not be used from inside of [DeepRecursiveScope] as it uses the call stack slot for initial recursive invocation. From inside of [DeepRecursiveScope] use
     * [callRecursive]
     * [DeepRecursiveScope.callRecursive].
     */
    public abstract suspend fun callRecursive(value: T): R

    /**
     * A scope class for [DeepRecursiveFunction] function declaration that defines [callRecursive] methods to recursively call this function or another [DeepRecursiveFunction] putting the call activation frame on the heap.
     */
    @RestrictsSuspension
    @SinceKotlin("1.4")
    @ExperimentalStdlibApi
    public sealed class DeepRecursiveScope<T, R> {
        /**
         * Makes recursive call to this [DeepRecursiveFunction] function putting the call activation frame on the heap, as opposed to the actual call stack that is used by a regular recursive call.
         */
        public abstract suspend fun callRecursive(value: T): R

        /**
         * Makes call to the specified [DeepRecursiveFunction] function putting the call activation frame on the heap, as opposed to the actual call stack that is used by a regular call.
         */

        @Deprecated(
            level = DeprecationLevel.ERROR,
            message = "'invoke' should not be called from DeepRecursiveScope. Use 'callRecursive' to do recursion in the heap instead of the call stack.",
            replaceWith = ReplaceWith("this.callRecursive(value)")
        )

        @Suppress("UNUSED_PARAMETER")
        public operator fun DeepRecursiveFunction<*, *>.invoke(value: Any?): Nothing =
            throw UnsupportedOperationException("Should not be called from DeepRecursiveScope")
    }

    // ================== Implementation
    @ExperimentalStdlibApi
    private typealias DeepRecursiveFunctionBlock = suspend DeepRecursiveScope<*, *>.(Any?) -> Any?

    private val UNDEFINED_RESULT = Result.success(COROUTINE_SUSPENDED)

    @Suppress("UNCHECKED_CAST")
    @ExperimentalStdlibApi
    private class DeepRecursiveScopeImpl<T, R>(
        block: suspend DeepRecursiveScope<T, R>.(T) -> R,

        value: T
    ) : DeepRecursiveScope<T, R>(), Continuation<R> {
        // Active function block
        private var function: DeepRecursiveFunctionBlock = block as DeepRecursiveFunctionBlock

        // Value to call function with
        private var value: Any? = value

        // Continuation of the current call
        private var cont: Continuation<Any?>? = this as Continuation<Any?>

        // Completion result (completion of the whole call stack)
        private var result: Result<Any?> = UNDEFINED_RESULT

        override suspend fun callRecursive(value: T): R =
            suspendCoroutineUninterceptedOrReturn {
                this.cont = it as Continuation<Any?>
                this.value = value
                COROUTINE_SUSPENDED
            }

    }
}
suspendCoroutineUninterceptedOrReturn { cont ->
    // calling another recursive function
    val function = block as DeepRecursiveFunctionBlock
    with(this@DeepRecursiveScopeImpl) {
        val currentFunction
        if (function != currentFunction) {
            // calling a different function -- create a
            trampoline to restore function ref
            this.function = function
            this.cont = crossFunctionCompletion(currentFunction, cont as Continuation<Any?>)
        } else {
            // calling the same function -- direct
            this.cont = this as Continuation<Any?>
        }
    }
    // When going back from a trampoline we cannot just call cont.resume (stack usage!)
    // We delegate the cont.resumeWith(it) call to runCallLoop
    cont = cont
    this.result = it
    @Suppress("UNCHECKED_CAST")
    fun runCallLoop(): R {
        while (true) {
            // This is block.startCoroutine(this, value, cont)
            function.startCoroutineUninterceptedOrReturn(1, value, cont)
            /*
             * Copyright 2010-2018
             * Copyright 2010-2018
             * Use of this source code is governed by the
             * Use of this source code is governed by the
             * Apache 2.0 license that can be found in the
             * Apache 2.0 license that can be found in the
             * license/LICENSE.txt file.
             * license/LICENSE.txt file.
             * Contributors.
             * Contributors.
             */

        /*
         * Copyright 2010-2018
         * Copyright 2010-2018
         */

        kotlin.internal.InlineOnly
        @SinceKotlin("1.3")
        @InlineOnly
        public inline fun Any?.hashCode(): Int = 
        @see Any.hashCode
        return (result as Result<R>).getOrThrow() // done -- final result
        // The order of comparison is important
    }
}

suspendCoroutineUninterceptedOrReturn { cont ->
    // calling another recursive function
    val function = block as DeepRecursiveFunctionBlock
    with(this@DeepRecursiveScopeImpl) {
        val currentFunction
        if (function != currentFunction) {
            // calling a different function -- create a
            trampoline to restore function ref
            this.function = function
            this.cont = crossFunctionCompletion(currentFunction, cont as Continuation<Any?>)
        } else {
            // calling the same function -- direct
            this.cont = this as Continuation<Any?>
        }
    }
    // When going back from a trampoline we cannot just call cont.resume (stack usage!)
    // We delegate the cont.resumeWith(it) call to runCallLoop
    cont = cont
    this.result = it
    @Suppress("UNCHECKED_CAST")
    fun runCallLoop(): R {
        while (true) {
            // This is block.startCoroutine(this, value, cont)
            function.startCoroutineUninterceptedOrReturn(1, value, cont)
            /*
             * Copyright 2010-2018
             * Copyright 2010-2018
             * Use of this source code is governed by the
             * Use of this source code is governed by the
             * Apache 2.0 license that can be found in the
             * Apache 2.0 license that can be found in the
             * license/LICENSE.txt file.
             * license/LICENSE.txt file.
             * Contributors.
             * Contributors.
             */

        /*
         * Copyright 2010-2018
         * Copyright 2010-2018
         */

        kotlin.internal.InlineOnly
        @SinceKotlin("1.3")
        @InlineOnly
        public inline fun Any?.hashCode(): Int = 
        @see Any.hashCode
        return (result as Result<R>).getOrThrow() // done -- final result
        // The order of comparison is important
    }
}

suspendCoroutineUninterceptedOrReturn { cont ->
    // calling another recursive function
    val function = block as DeepRecursiveFunctionBlock
    with(this@DeepRecursiveScopeImpl) {
        val currentFunction
        if (function != currentFunction) {
            // calling a different function -- create a
            trampoline to restore function ref
            this.function = function
            this.cont = crossFunctionCompletion(currentFunction, cont as Continuation<Any?>)
        } else {
            // calling the same function -- direct
            this.cont = this as Continuation<Any?>
        }
    }
    // When going back from a trampoline we cannot just call cont.resume (stack usage!)
    // We delegate the cont.resumeWith(it) call to runCallLoop
    cont = cont
    this.result = it
    @Suppress("UNCHECKED_CAST")
    fun runCallLoop(): R {
        while (true) {
            // This is block.startCoroutine(this, value, cont)
            function.startCoroutineUninterceptedOrReturn(1, value, cont)
            /*
             * Copyright 2010-2018
             * Copyright 2010-2018
             * Use of this source code is governed by the
             * Use of this source code is governed by the
             * Apache 2.0 license that can be found in the
             * Apache 2.0 license that can be found in the
             * license/LICENSE.txt file.
             * license/LICENSE.txt file.
             * Contributors.
             * Contributors.
             */

        /*
         * Copyright 2010-2018
         * Copyright 2010-2018
         */

        kotlin.internal.InlineOnly
        @SinceKotlin("1.3")
        @InlineOnly
        public inline fun Any?.hashCode(): Int = 
        @see Any.hashCode
        return (result as Result<R>).getOrThrow() // done -- final result
        // The order of comparison is important
    }
}
Open Source Used In DNAC Platform Arica 8308
Counts the number of set bits in the binary representation of this [Int] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Int.countOneBits(): Int
```

Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Int.countLeadingZeroBits(): Int
```

Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Int.countTrailingZeroBits(): Int
```

Returns a number having a single bit set in the position of the most significant set bit of this [Int] number, \n * or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Int.takeHighestOneBit(): Int
```

Returns a number having a single bit set in the position of the least significant set bit of this [Int] number, \n * or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Int.takeLowestOneBit(): Int
```

Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.rotateLeft(bitCount: Int): Int
```

Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.rotateRight(bitCount: Int): Int
```

Counts the number of set bits in the binary representation of this [Long] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Long.countOneBits(): Int
```

Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Long.countLeadingZeroBits(): Int
```

Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Long.countTrailingZeroBits(): Int
```

Returns a number having a single bit set in the position of the most significant set bit of this [Long] number, \n * or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Long.takeHighestOneBit(): Long
```

Returns a number having a single bit set in the position of the least significant set bit of this [Long] number, \n * or zero, if this number is zero.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public expect fun Long.takeLowestOneBit(): Long
```

Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateLeft(bitCount: Int): Long
```

Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateRight(bitCount: Int): Long
```
as the most significant bits on the left side.

* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:

`number.rotateRight(-n) == number.rotateLeft(n)`

* Rotating by a multiple of `Long.SIZE_BITS` (64) returns the same number, or more generally:

`number.rotateRight(n % 64) == number.rotateRight(n)`

> `number.rotateLeft(n) == number.rotateLeft(n % 64)`

 /*
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
 * `number.rotateRight(-n) == number.rotateLeft(n)`
 *
 * Rotating by a multiple of `Long.SIZE_BITS` (64) returns the same number, or more generally:
 * `number.rotateRight(n % 64) == number.rotateRight(n)`
 */

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateRight(bitCount: Int): Long

/**
 * Counts the number of set bits in the binary representation of this [Byte] number.
 */

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.countOneBits(): Int = (toInt() and 0xFF).countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Byte] number.
 */

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.countLeadingZeroBits(): Int =
    (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Byte] number.
 */

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number.
 */

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.takeHighestOneBit(): Byte = (toInt() and 0xFF).takeHighestOneBit().toByte()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Byte] number.
 */

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun Byte.takeLowestOneBit(): Byte = toInt().takeLowestOneBit().toByte()
number is zero.

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Represents a triad of values
 * There is no meaning attached to values in this class, it can be used for any purpose. An * Triplet type exhibits value semantics, i.e. two triples are equal if all three components are equal.
 * An * Example of decomposing it into values:
 * An * @sample sample.samples.misc.Triplet.triplesDestructuring
 */
public data class Triplet<out A, out B, out C>(
    public val first: A,
    public val second: B,
    public val third: C
) : Serializable

/**
 * Represents a triad of values
 * There is no meaning attached to values in this class, it can be used for any purpose. An * Triplet type exhibits value semantics, i.e. two triples are equal if all three components are equal.
 * An * Example of decomposing it into values:
 * An * @sample sample.samples.misc.Triplet.triplesDestructuring
 */
public data class Triplet<out A, out B, out C>(
    public val first: A,
    public val second: B,
    public val third: C
) : Serializable
/**
 * Returns string representation of the [Triple] including its [first], [second] and [third] values.
 */
public override fun toString(): String = "($first, $second, $third)"

/**
 * Converts this triple into a list.
 */
val List<T> = listOf(first, second, third)

/**
 * Converts this triple into a list.
 */
public fun <T> Triple<T, T, T>.toList(): List<T> =

/**
 * A range of values of type `UInt`. 
 */
@SinceKotlin("1.3")
public class UIntRange(start: UInt, endInclusive: UInt) : UIntProgression(start, endInclusive, 1), ClosedRange<UInt> {
    override val start: UInt get() = first
    override val endInclusive: UInt get() = last
    override fun contains(value: UInt): Boolean = first <= value && value <= last
    override fun isEmpty(): Boolean = first > last
    override fun equals(other: Any?): Boolean =
        other is UIntRange && (isEmpty() && other.isEmpty() ||
        first == other.first && last == other.last)
    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())
    override fun toString(): String = "$first..$last"
    companion object {
        /**
         * An empty range of values of type UInt. */
        public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)
    }
}

/**
 * A progression of values of type `UInt`. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public open class UIntProgression
    internal constructor(
        start: UInt,
        endInclusive: UInt,
        step: Int
    ) : Iterable<UInt> {
    init {
        if (step == 0.toInt()) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on negation."

    /**
     * The first element in the progression.
     */
    public val first: UInt = start

    /**
     * The last element in the progression.
     */
    public val last: UInt = getProgressionLastElement(start, endInclusive, step)

    /**
     * The step of the progression.
     */
    public val step: Int = step

    override fun iterator(): UIntIterator = UIntProgressionIterator(first, last, step)

    /**
     * Checks if the progression is empty.
     */
    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    companion object {
        /**
         * Creates UIntProgression within the specified bounds of a closed range.

         * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the
         * specified [step].
         */
        public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression = UIntProgression(rangeStart, rangeEnd, step)
    }
}

/**
 * An iterator over a progression of values of type `UInt`. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class UIntProgressionIterator(first: UInt, last: UInt, step: Int) : UIntIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private val step = step.toUInt() // use 2-complement math for negative steps
    private var next = if (hasNext) first else finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextUInt(): UInt {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        } else {
            next += step
        }
        return value
    }
}

/**
 * An iterator over a sequence of values of type `UByte`. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UByteIterator : Iterator<UByte> {
    override final fun next() = nextUByte()

    /**
     * Returns the next value in the sequence without boxing.
     */
    public abstract fun nextUByte(): UByte
}

/**
 * An iterator over a sequence of values of type `UShort`. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UShortIterator : Iterator<UShort>
override final fun next() = nextUShort() /* Returns the next value in the sequence without boxing. */

public abstract fun nextUShort(): UShort

/**
 * Returns the next value in the sequence without boxing.
 */

public abstract fun nextULong(): ULong

/**
 * An iterator over a sequence of values of type 'ULong'.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class ULongIterator : Iterator<ULong>

override final fun next() = nextULong() /* Returns the next value in the sequence without boxing. */

/**
 * An iterator over a sequence of values of type 'ULong'.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class ULongProgression
internal constructor(
    start: ULong,
    endInclusive: ULong,
    step: Long
) : Iterable<ULong> {
    init {
        if (step == 0) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Long.MIN_VALUE to avoid overflow on negation.")
    }

    /**
     * The first element in the progression.
     */
    public val first: ULong = start

    /**
     * The last element in the progression.
     */
    public val last: ULong

    /**
     * The step of the progression.
     */
    public val step: Long = step

    override fun iterator(): ULongIterator = ULongProgressionIterator(first, last, step)

    /** Checks if the progression is empty. */
    @SinceKotlin("1.3")
    public fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    companion object {
        /**
         * Creates ULongProgression within the specified bounds of a closed range.
         * In order to go backwards the [step] must be negative.
         */
        public fun fromClosedRange(rangeStart: ULong, rangeEnd: ULong, step: Long): ULongProgression = ULongProgression(rangeStart, rangeEnd, step)

        /**
         * A progression of values of type 'ULong'.
         */
        @SinceKotlin("1.3")
        @ExperimentalUnsignedTypes
        public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1), ClosedRange<ULong> {

            override val start: ULong get() = first
            override val endInclusive: ULong get() = last

            override fun contains(value: ULong): Boolean = first <= value && value <= last

            override fun isEmpty(): Boolean = first > last

            override fun equals(other: Any?): Boolean =
                other is ULongRange && (isEmpty() && other.isEmpty() ||
                        first == other.first && last == other.last)

            override fun hashCode(): Int =
                if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())

            override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $-step"

            companion object {
                /**
                 * An empty range of values of type ULong.
                 */

                public val EMPTY: ULongRange = ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)
            }
        }
    }

    /**
     * An iterator over a progression of values of type 'ULong'.
     */
    @SinceKotlin("1.3")
    @ExperimentalUnsignedTypes
    private class ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {
        private val finalElement = last
        private var hasNext: Boolean = if (step > 0) first <= last else first >= last
        private val step = step.toULong() // use 2-complement math for negative steps

        override fun hasNext(): Boolean = hasNext

        override fun nextULong(): ULong {
            val value = next
            if (value == finalElement) {
                if (!hasNext) throw kotlin.NoSuchElementException()
                hasNext = false
            } else {
                next += step
            }
            return value
        }
    }
}


by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
package kotlin.math

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: UInt, b: UInt): UInt {
    return minOf(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: ULong, b: ULong): ULong {
    return minOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: UInt, b: UInt): UInt {
    return maxOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: ULong, b: ULong): ULong {
    return maxOf(a, b)
}

/**
 * Counts the number of set bits in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countOneBits(): Int = toInt().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant bit of this [UInt] number, or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeHighestOneBit(): UInt = toInt().takeHighestOneBit().toUInt()

/**
 * Returns a number having a single bit set in the position of the least significant bit of this [UInt] number, or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeLowestOneBit(): UInt = toInt().takeLowestOneBit().toUInt()

/**
 * Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateLeft(bitCount: Int): UInt = toInt().rotateLeft(bitCount).toUInt()

/**
 * Rotates the binary representation of this [UInt] number right by the specified [bitCount] number of bits.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt()

rw package kotlin.math

*/

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("UNumbersKt")
package kotlin

/**
 * Counts the number of set bits in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countOneBits(): Int = toLong().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant bit of this [ULong] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeHighestOneBit(): ULong = toInt().takeHighestOneBit().toUInt()

/**
 * Returns a number having a single bit set in the position of the least significant bit of this [ULong] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeLowestOneBit(): ULong = toInt().takeLowestOneBit().toUInt()

/**
 * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.rotateLeft(bitCount: Int): ULong = toInt().rotateLeft(bitCount).toUInt()

/**
 * Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.rotateRight(bitCount: Int): ULong = toInt().rotateRight(bitCount).toUInt()

```

Open Source Used In DNAC Platform Arica 8314
Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULong.countLeadingZeroBits(): Int =
toLong().countLeadingZeroBits()  // Counts the number of consecutive least significant bits that are zero in
// the binary representation of this [ULong] number.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULong.countTrailingZeroBits(): Int =
toLong().countTrailingZeroBits()  // Returns a number having a single bit set in the position of the most
// significant set bit of this [ULong] number,\n// or zero, if this number is zero.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULong.takeHighestOneBit(): ULong =
toLong().takeHighestOneBit();  // Rotates the binary representation of this [ULong] number left
// by the specified [bitCount] number of bits.\n// The most significant bits pushed out from the left side reenter
// the number as the least significant bits on the right side.\n// Rotating the number left by a negative bit count is the
// same as rotating it right by the negated bit count:\n// Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\n
`\nnumber.rotateLeft(n) == number.rotateLeft(n % 64)`\n
/*@SinceKotlin("1.3")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULong.rotateLeft(bitCount: Int): ULong =
toLong().rotateLeft(bitCount);  // Rotates the binary representation of this [ULong] number right by the specified
// [bitCount] number of bits.\n// The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n// Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n// Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally\n
`\nnumber.rotateRight(-n) == number.rotateLeft(n)`\n
/*@SinceKotlin("1.3")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULong.rotateRight(bitCount: Int): ULong =
toLong().rotateRight(bitCount);  // Counts the number of set bits in the binary representation of this [UByte] number.\n
/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.countOneBits(): Int =
toUInt().countOneBits();  // Counts the number of consecutive least significant bits that are zero in
// the binary representation of this [UByte] number.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.countLeadingZeroBits(): Int =
toByte().countLeadingZeroBits()  // Counts the number of consecutive least significant bits that are zero in
// the binary representation of this [UByte] number.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.countTrailingZeroBits(): Int =
toByte().countTrailingZeroBits()  // Returns a number having a single bit set in the position of the most
// significant set bit of this [UByte] number,\n// or zero, if this number is zero.

/*@SinceKotlin("1.4")*/@WasExperimental(ExperimentalStdlibApi::class)@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.takeHighestOneBit(): UByte =
toInt().takeHighestOneBit();  // Returns a number having a single bit set in the position of the least
// significant set bit of this [UByte] number,\n// or zero, if this number is zero.\n
Open Source Used In DNAC Platform Arica 8315
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.takeLowestOneBit(): UByte =
toInt().takeLowestOneBit().toUByte()

`\n
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateLeft(bitCount: Int): UByte =
toByte().rotateLeft(bitCount).toUByte()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateRight(bitCount: Int): UByte =
toByte().rotateRight(bitCount).toUByte()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.takeHighestOneBit(): UShort =
toInt().takeHighestOneBit().toUShort()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.takeLowestOneBit(): UShort =
toInt().takeLowestOneBit().toUShort()

`\n
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.rotateLeft(bitCount: Int): UShort =
toShort().rotateLeft(bitCount).toUShort()

`\n
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.rotateRight(bitCount: Int): UShort =
toShort().rotateRight(bitCount).toUShort()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.countLeadingZeroBits(): Int =
toInt().countLeadingZeroBits()

`\n
@SinceKotlin("1.4")
@WasExperimental( ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.countTrailingZeroBits(): Int =
toInt().countTrailingZeroBits()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.takeLowestOneBit(): UByte =
toInt().takeLowestOneBit().toUByte()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.rotateLeft(bitCount: Int): UByte =
toByte().rotateLeft(bitCount).toUByte()

`\n
*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.rotateRight(bitCount: Int): UByte =
toByte().rotateRight(bitCount).toUByte()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.countOneBits(): Int =
toUInt().countOneBits()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.countLeadingZeroBits(): Int =
toShort().countLeadingZeroBits()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.countTrailingZeroBits(): Int =
toShort().countTrailingZeroBits()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.takeHighestOneBit(): UShort =
toInt().takeHighestOneBit().toUShort()

`\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShort.takeLowestOneBit(): UShort =
toInt().takeLowestOneBit().toUShort()

`\n
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateLeft(bitCount: Int): UByte =
toByte().rotateLeft(bitCount).toUByte()

`\n
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateRight(bitCount: Int): UByte =
toByte().rotateRight(bitCount).toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.takeLowestOneBit(): UByte =
toInt().takeLowestOneBit().toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateLeft(bitCount: Int): UByte =
toByte().rotateLeft(bitCount).toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateRight(bitCount: Int): UByte =
toByte().rotateRight(bitCount).toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.takeLowestOneBit(): UByte =
toInt().takeLowestOneBit().toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateLeft(bitCount: Int): UByte =
toByte().rotateLeft(bitCount).toUByte()

`\n
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByte.rotateRight(bitCount: Int): UByte =
toByte().rotateRight(bitCount).toUByte()
returns the same number, or more generally `n * `number.rotateRight(n) == number.rotateRight(n % 16)`

```kotlin
*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
\n\npublic inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()*/

```
representation of a number.

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)

 /**
 * Parses the string as a signed [UByte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByteOrNull(): UByte? {
    val int = this.toUIntOrNull() ?: return null
    if (int <= UByte.MAX_VALUE) return int.toUByte()
    return null
}

/**
 * Parses the string as a signed [UShort] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShort(): UShort = toUShortOrNull() ?: numberFormatError(this)

 /**
 * Parses the string as a signed [UShort] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShortOrNull(radix: Int): UShort? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int <= UShort.MAX_VALUE) return int.toUShort()
    return null
}

/**
 * Parses the string as an [UInt] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUInt(): UInt = toUIntOrNull() ?: numberFormatError(this)

 /**
 * Parses the string as an [UInt] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUIntOrNull(radix: Int): UInt? {
    checkRadix(radix)
    val length = this.length
    if (length == 0) return null

    val limit: UInt = UInt.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar != '-')
        return null

    var pos = length
    if (firstChar == '+')
        pos--
    else if (firstChar == '-')
        pos--

    var sign = true
    if (firstChar == '+')
        sign = false
    else if (firstChar == '-')
        sign = false

    while (pos > 0 && this[pos] in '0'..'9')
        pos--

    return when (sign)
    }

*/
fun toULongOrNull(radix: Int): ULong? {
    checkRadix(radix)
    val length = this.length
    if (length == 0) return null
    val limit: ULong = ULong.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 512409557603043100uL  // limit / 36
    val uradix = radix.toULong()
    var result = 0uL
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / uradix
                if (result > limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }

        result *= uradix

        val beforeAdding = result
        result += digit.toUInt()
        if (result < beforeAdding) return null // overflow has happened
    }

    return result
}

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(): ULong? = toULongOrNull(radix = 10)

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(radix: Int): ULong? {
    checkRadix(radix)
    val length = this.length
    if (length == 0) return null
    val limit: ULong = ULong.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 512409557603043100uL  // limit / 36
    val uradix = radix.toULong()
    var result = 0uL
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / uradix
                if (result > limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }

        result *= uradix

        val beforeAdding = result
        result += digit.toUInt()
        if (result < beforeAdding) return null // overflow has happened
    }

    return result
}
sinceKotlin("1.2")

/** Computes the arc cosine of the value [x];\n * the returned value is an angle in the range from `0.0` to `\PI` radians.\n */
@SinceKotlin("1.2")
public expect fun acos(x: Double): Double

/** Computes the tangent of the angle \([x]\) given in radians.\n */
@SinceKotlin("1.2")
public expect fun tan(x: Double): Double

/** Computes the sine of the value \([x]\);\n * the returned value is an angle in the range from `-\PI/2` to `\PI/2` radians.\n */
@SinceKotlin("1.2")
public expect fun sin(x: Double): Double

/** Computes Euler's number \(e\).\n */
@SinceKotlin("1.2")
public expect fun exp(x: Double): Double

/** Computes the square root of the value \([x]\);\n * the returned value is an angle in the range from `0.0` to \(\PI\) radians.\n */
@SinceKotlin("1.2")
public expect fun sqrt(x: Double): Double
natural logarithm (base 'E') of the value [x].\n
Special cases:\n
- `ln(NaN)` is `NaN`
- `ln(x)` is `NaN` when `x < 0.0` or `- ln(+Inf)` is `+Inf`
- `- ln(0.0)` is `-Inf`

This function can be implemented to produce more precise
result for [x] near zero.\n
Special cases:\n
- `- ln1p(NaN)` is `NaN`
- `- ln1p(x)` is `NaN` where `x < -1.0` or `- ln1p(-1.0)` is `-Inf`
- `- ln1p(+Inf)` is `+Inf`

This function returns the smallest double value that is greater than or equal
to the given value [x] and is a mathematical integer.\n
This function can be implemented to produce more precise
result for [x] near zero.\n
Special cases:\n
- `ceil(x)` is `x` where `x` is `NaN` or `-Inf` or `-Inf` or already a mathematical integer\n- `floor(x)` is `x` where `x` is `NaN` or `-Inf` or `-Inf` or already a mathematical integer\n
This function returns the smallest of two values.\n
If either value is `NaN`, then the result is `NaN`.\n
This function returns the absolute value of this value.\n
This function returns the sign of this value:
- `-1.0` if the value is negative,
- zero if the value is zero,
- `1.0` if the value is positive

This function returns the ulp (unit in the last place) of this value.\n
This function returns the value nearest to this value in direction of positive infinity.\n
This function returns the [Double] value larger in magnitude\nand with the sign bit same as of the [sign] value.\n
If [sign] is `NaN` the sign of the result is undefined.\n
This function returns the ulp (unit in the last place) of this value.\n
An ulp is a positive distance between this value and the next nearest [Double] value larger in magnitude.\n
This function returns the [Double] value nearest to this value in direction of positive infinity.\n
This function returns the [Double] value nearest to this value in direction of positive infinity.\n
This function returns the [Double] value nearest to this value in direction of positive infinity.\n
This function returns the [Double] value nearest to this value in direction of positive infinity.
Double

```kotlin
/**
 * Returns the [Double] value nearest to this value in direction of negative infinity.
 */
public expect fun Double.nextDown(): Double

/**
 * Returns the [Double] value nearest to this value in direction from this value towards the value [to].
 * Special cases:
 * - `x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`
 * - `x.nextTowards(x) == x`
 */
@SinceKotlin("1.2")
public expect fun Double.nextTowards(to: Double): Double

/**
 * Rounds this [Double] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 */
@SinceKotlin("1.2")
public expect fun Double.roundToInt(): Int

/**
 * Computes the sine of the angle [x] in radians.
 */
@SinceKotlin("1.2")
public expect fun sin(x: Float): Float

/**
 * Computes the cosine of the angle [x] in radians.
 */
@SinceKotlin("1.2")
public expect fun cos(x: Float): Float

/**
 * Computes the tangent of the angle [x] in radians.
 */
@SinceKotlin("1.2")
public expect fun tan(x: Float): Float

/**
 * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x].
 */
@SinceKotlin("1.2")
public expect fun atan2(y: Float, x: Float): Float

/**
 * Computes the hyperbolic sine of the value [x].
 */
@SinceKotlin("1.2")
public expect fun sinh(x: Float): Float

/**
 * Computes the hyperbolic cosine of the value [x].
 */
@SinceKotlin("1.2")
public expect fun cosh(x: Float): Float

/**
 * Computes the hyperbolic tangent of the value [x].
 */
@SinceKotlin("1.2")
public expect fun tanh(x: Float): Float

```
`tanh(NaN)` is `NaN` when `x > 1` or `x < -1` or `-tanh(1.0)` is `-Inf` * `tanh(-1.0)` is `-Inf` * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow. * Special cases: * - returns `-Inf` if any of arguments is infinite or returns `NaN` if any of arguments is `NaN` and the other is not infinite

* * @public fun hypot(x: Float, y: Float): Float

/** Computes the square root of the value [x], [y] * Special cases: * - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`
 * @public fun sqrt(x: Float): Float

/** Computes Euler's number `e` raised to the power of the value [x], [y] * Special cases: * - `exp(NaN)` is `NaN` * - `exp(+Inf)` is `+Inf` * - `exp(-Inf)` is `0.0` * Computes `exp(x)` - `I` * * This function can be implemented to produce more precise result for [x] near zero. * * Special cases: * - `expm1(NaN)` is `NaN` * - `expm1(+Inf)` is `+Inf` * - `expm1(-Inf)` is `-1.0` * * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].

* * @public fun ln1p(x: Float): Float

/** Computes the binary logarithm (base 2) of the value [x], [y] * See [ln] function for special cases. * * @public fun log(x: Float, base: Float): Float

/** Computes Euler's number `e` raised to the power of the value [x]. * Special cases: * - `exp(NaN)` is `NaN` * - `exp(+Inf)` is `+Inf` * - `exp(-Inf)` is `-Inf` * * Computes `exp(x)` - `1` * * This function can be implemented to produce more precise result for [x] near zero. * * Special cases: * - `expm1(NaN)` is `NaN` * - `expm1(+Inf)` is `+Inf` * - `expm1(-Inf)` is `-Inf` * * See also logarithm functions for special fixed bases: [ln], [log10] and [log2].

* * @public fun exp(x: Float): Float

/** Computes `exp(x) - 1`.

* * @public fun expm1(x: Float): Float

/** Computes the logarithm of the value [x] to the given [base]. * Special cases: * - `log(x, b)` is `NaN` if either `x` or `b` are `NaN` * - `log(x, b)` is `NaN` when `x < 0` or `b <= 0` or `b == 1.0` * - `log(+Inf, +Inf)` is `NaN` * - `log(+Inf, b)` is `-Inf` for `b < 1` * - `log(0.0, b)` is `-Inf` for `b > 1` and `-Inf` for `b > 1` * - `log(x, b)` is `-Inf` when `x = 0.0` * - `log(x, b)` is `-Inf` when `x < 0` * - `log(x, b)` is `-Inf` when `x < 0` * - `ln(0.0)` is `-Inf` * * Computes the common logarithm (base 10) of the value [x], [y] * See [ln] function for special cases, [log10], [exp],...
`NaN` for `x != 0.0` * `-b.pow(Inf)` is `NaN` for `abs(b) == 1.0` * `-b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer

* `b.pow(Inf)` is `NaN` for `abs(b) == 1.0`
* `b.pow(x)` is `NaN` for `b < 0` and `x` is finite and not an integer

```kotlin
@SinceKotlin("1.2")
public expect fun Float.pow(x: Float): Float

/**
 * Raises this value to the integer power [n].
 * Special cases:
 *   - `NaN.pow(0)` is `NaN`
 *   - `NaN.abs*` is `NaN`
 * @see abs function

@SinceKotlin("1.2")
public expect val Float.abs: Float

/**
 * Returns the absolute value of this value.
 * Special cases:
 *   - `NaN.abs*` is `NaN`
 * @see abs function

@SinceKotlin("1.2")
public expect val Float.sign: Float

/**
 * Returns this value with the sign bit same as of the [sign] value.
 * Special cases:
 *   - If [sign] is `NaN` the sign of the result is undefined.
 *   - `NaN.sign*` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Float): Float

/**
 * Returns this value with the sign bit same as of the [sign] value.
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Int): Float

/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 * Special cases:
 *   - `Int.MIN_VALUE` when `x > Int.MAX_VALUE`
 *   - `x.roundToLong()` when `x < Int.MIN_VALUE`
   * @throws IllegalArgumentException when this value is `NaN`
 */
@SinceKotlin("1.2")
public expect fun Float.roundToInt(): Int

/**
 * Rounds this [Float] value to the nearest integer and converts the result to
 * [Long].
 * Special cases:
 *   - `Long.MIN_VALUE` when `x < Long.MAX_VALUE`
 *   - `x.roundToLong()` when `x > Long.MIN_VALUE`
   * @throws IllegalArgumentException when this value is `NaN`
 */
@SinceKotlin("1.2")
public expect fun Float.roundToLong(): Long
```

Open Source Used In DNAC Platform Arica  8324
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica 8376
Open Source Used in DNAC Platform Arica 8466
Open Source Used In DNAC Platform Arica
Open Source Used in DNAC Platform Arica 8500
Open Source Used In DNAC Platform Arica

8510

Y;MAAC,OAAM,WA66P3,B,YAACL,CK;:IDSM5If,p,M;BAA6C,OAAM,SAAP,WAAO,EAAS,OAAO,CA47;:S6,CAD
pDo,B;MAASc,c;:QA60D,D;QA60D,B;IAAL,bzKhD,Y;QA60D,D;lb6xKzD,Y;QA60D,C;UAAY,B;yB;UAAM,IAAI,Ch1KD,d;B,0f
DvC,Ob1KDc,D,Ca0KDc,D,C;YAAAY,B;aaAO,K;YYAP,c;:QAAC,c;CAAO,A;I;:Mb3K3Do,B;iK;:CA67,D,Y;MAAU,C,O
AAO,qBAAP,WAAO,CK;:0C3C3,Y;MAC+C,gBAAP,W;MAAA,OAAM,CAAA,C;QAC,0C3C3,C;SDKuOgC,GxUPc,C,SN
o4Bo,B;Q;::;IA3b,C5,B;MAlc,B;IAEC,4B;MAAQ,D;OAAl,QAAS,OAAT,GAAG,B;CAAP,B;GAAG,CAAT,
QAAS,CAAhC,GAA8C,W,K;FM;AEnG,yB;MAAA,qD;MAAA,OAAM,MAAl,bK;O;KAIL1C,c;+FAAO,yB;MAAA,
+:D;MAAA,mB;QMwD,ubO;KANX,c;2FAAQ,yB;MAAA,:+D;MAAA,mB;QMAD,uO;KANP,c;CIAAQ,
+mC;MAKlOAAl,QAAS,OAAO,KAAB,C;CAAb,GAAwB,gBAAXB,GAAYc,B;IAAAuB,SBAAk,B;QAIAIB,EAwC,1
IAAC,C;CAAC,V,C;K;IAE7,C;QAAS,OAAT,OAAT,KAAb,B;CAAb,GAAwB,GAAB,B;CAAb,B,SBAAk,B,SBAAk,
QAIAIB,EAwC,IAAC,V;CAAC,V;K;IAE7,C;CAAV,C;IAAC,V,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,OAAT,QAAS,
Open Source Used In DNAC Platform Arica 8519
Open Source Used In DNAC Platform Arica 8522
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica
**1.1800 define-property 0.2.5 0.2.5**

**1.1800.1 Available under license:**

The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1801 negotiator 0.5.3 0.5.3

1.1801.1 Available under license:

(The MIT License)

Copyright (c) 2012-2014 Federico Romero
Copyright (c) 2012-2014 Isaac Z. Schlueter
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1802 commons-el 5.5.23

1.1802.1 Available under license:

Apache Commons CLI
Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1803 assign-symbols 1.0.0 1.0.0

1.1803.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1804 regex-cache 0.4.4

1.1804.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1805 jna 5.8.0

1.1805.1 Available under license:

/*
 * The contents of this file are subject to the Mozilla Public License Version 1.1
 * (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at http://www.mozilla.org/MPL/
 *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the License.
 *
 * The Original Code is 'iText, a free JAVA-PDF library'.
 *
 * The Initial Developer of the Original Code is Bruno Lowagie. Portions created by
 * the Initial Developer are Copyright (C) 1999, 2000, 2001, 2002 by Bruno Lowagie.
 * All Rights Reserved.
 * Co-Developer of the code is Paulo Soares. Portions created by the Co-Developer
 * are Copyright (C) 2000, 2001, 2002 by Paulo Soares. All Rights Reserved.
 *
 * Contributor(s): all the names of the contributors are added in the source code
 * where applicable.
 *
 * Alternatively, the contents of this file may be used under the terms of the
 * LGPL license (the "GNU LIBRARY GENERAL PUBLIC LICENSE"), in which case the
 * provisions of LGPL are applicable instead of those above. If you wish to
 * allow use of your version of this file only under the terms of the LGPL
 * License and not to allow others to use your version of this file under
 * the MPL, indicate your decision by deleting the provisions above and
 * replace them with the notice and other provisions required by the LGPL.
 * If you do not delete the provisions above, a recipient may use your version
 * of this file under either the MPL or the GNU LIBRARY GENERAL PUBLIC LICENSE.
 *
 * This library is free software; you can redistribute it and/or modify it
 * under the terms of the MPL as stated above or under the terms of the GNU
 * Library General Public License as published by the Free Software Foundation;
 * either version 2 of the License, or any later version.
 *
 * This library is distributed in the hope that it will be useful, but WITHOUT
 * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
 * FOR A PARTICULAR PURPOSE. See the GNU Library general Public License for more
 * details.
* If you didn't download this code from the following link, you should check if
* you aren't using an obsolete version:
* http://www.lowagie.com/iText/
*
* This class is generated based on a grammar file privided by SUN, and updated
* by Carsten Hammer. SUN's license agreement can be found at this URL:
* http://java.sun.com/products/java-media/2D/samples/samples-license.html
* See also the file sun.txt in directory com.lowagie.text.pdf
*/

Maven Ant Tasks
Copyright 2002-2011 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
================================================================================================================================================================
================================================================================================================================================================

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed
their package names from 'org.jfree' to 'clover.org.jfree'. This was necessary to avoid potential name
conflicts during instrumentation of a code using the original library when using Clover. No source code of
the original library was modified.
================================================================================================================================================================
================================================================================================================================================================

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

   a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

   b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

   c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
That's all there is to it!

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.
Apache Ant
Copyright 1999-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark
Graphics Corp that has been kindly donated to the Apache Software
Foundation.

/*
 *
 *        Apache License
 *        Version 2.0, January 2004
 *        http://www.apache.org/licenses/
 *
 *   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
* to the Licensor or its representatives, including but not limited to
* communication on electronic mailing lists, source code control systems,
* and issue tracking systems that are managed by, or on behalf of, the
* Licensor for the purpose of discussing and improving the Work, but
* excluding communication that is conspicuously marked or otherwise
* designated in writing by the copyright owner as "Not a Contribution."

* "Contributor" shall mean Licensor and any individual or Legal Entity
* on behalf of whom a Contribution has been received by Licensor and
* subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* copyright license to reproduce, prepare Derivative Works of,
* publicly display, publicly perform, sublicense, and distribute the
* Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* (except as stated in this section) patent license to make, have made,
* use, offer to sell, sell, import, and otherwise transfer the Work,
* where such license applies only to those patent claims licensable
* by such Contributor that are necessarily infringed by their
* Contribution(s) alone or by combination of their Contribution(s)
* with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:

* (a) You must give any other recipients of the Work or
*     Derivative Works a copy of this License; and
*
* (b) You must cause any modified files to carry prominent notices
*     stating that You changed the files; and
*
* (c) You must retain, in the Source form of any Derivative Works
*     that You distribute, all copyright, patent, trademark, and
attribute notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribute notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribute
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribute notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
you any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement You may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriate use or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can
be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain.
SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
=================================================================

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the
Open Source Used In DNAC Platform Arica 8575

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Lang
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

/*
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2000-2003 The Apache Software Foundation. All rights reserved.
 * *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 * *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 * *
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 * *
 * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:
 * "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgement may appear in the software itself,
Open Source Used In DNAC Platform Arica 8587

* if and wherever such third-party acknowledgements normally appear.

* 4. The names "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache" nor may "Apache" appear in their names without prior written permission of the Apache Group.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* ==============================================================

* This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

*/

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

MOZILLA PUBLIC LICENSE
Version 1.1

---------

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor’s choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and
(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after
the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or
collateral in which You describe recipients’ rights relating to the
Covered Code. You may distribute the Executable version of Covered
Code or ownership rights under a license of Your choice, which may
contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the
Executable version does not attempt to limit or alter the recipient’s
rights in the Source Code version from the rights set forth in this
License. If You distribute the Executable version under a different
license You must make it absolutely clear that any terms which differ
from this License are offered by You alone, not by the Initial
Developer or any Contributor. You hereby agree to indemnify the
Initial Developer and every Contributor for any liability incurred by
the Initial Developer or such Contributor as a result of any such
terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code
not governed by the terms of this License and distribute the Larger
Work as a single product. In such a case, You must make sure the
requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Code due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description
must be included in the LEGAL file described in Section 3.4 and must
be included with all distributions of the Source Code. Except to the
extent prohibited by statute or regulation, such description must be
sufficiently detailed for a recipient of ordinary skill to be able to
understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has
attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised
and/or new versions of the License from time to time. Each version
will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the
License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:
(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY.
RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.


11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as
"Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

```
```

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is ____________________________.

The Initial Developer of the Original Code is ______________________.

Portions created by _______________________ are Copyright (C) ______ ____________________________. All Rights Reserved.

Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

/*

Copyright (c) 2000, Derek Petillo
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Praxis Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Public Domain Dedication
This license is acceptable for Free Cultural Works.

Copyright-Only Dedication (based on United States law) or Public Domain Certification

The person or persons who have associated work with this document (the “Dedicator” or “Certifier”) hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the “Work”) to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a “dedicator” below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.
Open Source License for the overlibmws Package

1. License coverage

Note that this license only covers the script library (javascript core and plugin modules) and not any supporting material such as the overlibmws website or its online documentation and support files. You may not reproduce the website or its online material without explicit written permission from the author, but can freely incorporate scripts and procedures which are demonstrated in that material into your own HTML or XML documents.

2. License (Artistic)

Preamble
The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:
"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.
You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.
You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as ftp.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

use the modified Package only within your corporation or organization.
rename any non-standard executables so the names do not conflict with standard executables, which must also
be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

make other distribution arrangements with the Copyright Holder.

You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

accompany the distribution with the machine-readable source of the Package with your modifications.

accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

make other distribution arrangements with the Copyright Holder.

You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own.

The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this Package.

C or perl subroutines supplied by you and linked into this Package shall not be considered part of this Package. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
=================================================================================================

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'it.unimi.dsi.fastutil' to 'clover.it.unimi.dsi.fastutil'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=================================================================================================

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".
A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice

That's all there is to it!

Javolution - Java(TM) Solution for Real-Time and Embedded Systems  
Copyright (c) 2006, Javolution (http://javolution.org)  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The libffi source distribution contains certain code that is not part of libffi, and is only used as tooling to assist with the building and testing of libffi. This includes the msvcc.sh script used to wrap the Microsoft compiler with GNU compatible command-line options, make_sunver.pl, and the libffi test code distributed in the testsuite/libffi.bhaible directory. This code is distributed with libffi for the purpose of convenience only, and libffi is in no way derived from this code.

msvcc.sh an testsuite/libffi.bhaible are both distributed under the terms of the GNU GPL version 2, as below.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a “work based on the Program” means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

libffi - Copyright (c) 1996-2020 Anthony Green, Red Hat, Inc and others. See source files for details.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Commons BeanUtils

Copyright 2000-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

=====================================================================

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from 'org.apache.velocity' to 'clover.org.apache.velocity'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

=====================================================================

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 $Id: LICENSE.txt,v 1.1.1.1 2004/07/01 13:59:13 jvanzyl Exp $

Copyright 2002 (C) The Codehaus. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name "classworlds" must not be used to endorse or promote products derived from this Software without prior written permission of The Codehaus. For written permission, please contact bob@codehaus.org.

4. Products derived from this Software may not be called "classworlds" nor may "classworlds" appear in their names without prior written permission of The Codehaus. "classworlds" is a registered trademark of The Codehaus.

5. Due credit should be given to The Codehaus. (http://classworlds.codehaus.org/).

THIS SOFTWARE IS PROVIDED BY THE CODEHAUS AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CODEHAUS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/
==================================================================================================

SOFTWARE RIGHTS

ANTLR 1989-2006 Developed by Terence Parr
Partially supported by University of San Francisco & jGuru.com

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'antlr.*' to the 'clover.antlr.*'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==================================================================================================

Open Source Used In DNAC Platform Arica 8631
We reserve no legal rights to the ANTLR--it is fully in the public domain. An individual or company may do whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs (commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

The primary ANTLR guy:

Terence Parr
parrt@cs.usfca.edu
parrt@antlr.org

==================================================================================

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.google.common' to 'clover.com.google.common'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==================================================================================

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==================================================================================

==========================

Modified by Atlassian
The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.google.json' to 'clover.com.google.json'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==================================================================================

==

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package name from 'com.lowagie' to 'clover.com.lowagie'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==================================================================================

=Mozilla Public License Version 1.1=

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.
1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
   A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.
   B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available.
for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an
unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients’ rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12)
months after the date it initially became available, or at least six
(6) months after a subsequent version of that particular Modification
has been made available to such recipients. You are responsible for
ensuring that the Source Code version remains available even if the
Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a
file documenting the changes You made to create that Covered Code and
the date of any change. You must include a prominent statement that
the Modification is derived, directly or indirectly, from Original
Code provided by the Initial Developer and including the name of the
Initial Developer in (a) the Source Code, and (b) in any notice in an
Executable version or related documentation in which You describe the
origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's
intellectual property rights is required to exercise the rights
granted by such Contributor under Sections 2.1 or 2.2,
Contributor must include a text file with the Source Code
distribution titled "LEGAL" which describes the claim and the
party making the claim in sufficient detail that a recipient will
know whom to contact. If Contributor obtains such knowledge after
the Modification is made available as described in Section 3.2,
Contributor shall promptly modify the LEGAL file in all copies
Contributor makes available thereafter and shall take other steps
(such as notifying appropriate mailing lists or newsgroups)
reasonably calculated to inform those who received the Covered
Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming
interface and Contributor has knowledge of patent licenses which
are reasonably necessary to implement that API, Contributor must
also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to
Section 3.4(a) above, Contributor believes that Contributor's
Modifications are Contributor's original creation(s) and/or
Contributor has sufficient rights to grant the rights conveyed by
this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source
Code. If it is not possible to put such notice in a particular Source
Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS.
WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant
alleging that such Participant’s Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.


11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions.
With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

```
The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is ________________________________.

The Initial Developer of the Original Code is _______________________.
Portions created by __________________ are Copyright (C) _____ ____________________________. All Rights Reserved.
```
Open Source Used In DNAC Platform Arica

Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [___] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License.”

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

Apache Velocity

Copyright (C) 2000-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright
   statements and notices. Redistributions must also contain a
   copy of this document.

2. Redistributions in binary form must reproduce the
   above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other
   materials provided with the distribution.

3. The name "DOM4J" must not be used to endorse or promote
   products derived from this Software without prior written
   permission of MetaStuff, Ltd. For written permission,
   please contact dom4j-info@metastuff.com.

Open Source Used In DNAC Platform Arica 8651
4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.

5. Due credit should be given to the DOM4J Project -
http://www.dom4j.org

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*--
$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp $

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the
end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

/*
From: http://www.json.org/license.html

=================================================================

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
SPDX-License-Identifier: Apache-2.0 OR LGPL-2.1

Java Native Access (JNA) is licensed under the LGPL, version 2.1
or later, or (from version 4.0 onward) the Apache License,
version 2.0.

You can freely decide which license you want to apply to the project.

You may obtain a copy of the LGPL License at:

http://www.gnu.org/licenses/licenses.html

A copy is also included in the downloadable source code package
containing JNA, in file "LGPL2.1", under the same directory
as this file.

You may obtain a copy of the Apache License at:

http://www.apache.org/licenses/

A copy is also included in the downloadable source code package
containing JNA, in file "AL2.0", under the same directory
as this file.

Commercial support may be available, please e-mail

----------------------------------------------------------------------------------

Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed
their package name from 'org.apache.commons' to 'clover.org.apache.commons'. This was necessary to
avoid potential name conflicts during instrumentation of a code using the original library when using Clover.
No source code of the original library was modified.

--------------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
* names of its contributors may be used to endorse or promote
* products derived from this software without specific prior
* written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

********************************************************************************/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

http://www.atlassian.com/dms/wac/Atlassian_EULA_4-2.pdf

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor’s protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors’ reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don’t assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that
they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

. Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
/*--
$Id: LICENSE.txt,v 1.11 2004/02/06 09:32:57 jhunter Exp $

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows
these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ''AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/
Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

   "This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana Univeristy" and "Indiana Univeristy Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana Univeristy" name nor may "Indiana Univeristy" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

==================================================================================
Modified by Atlassian

The binary file of the original library has been modified by Atlassian in such way that classes have changed their package names from the 'org.apache.commons' to the 'clover.org.apache.commons'. This was necessary to avoid potential name conflicts during instrumentation of a code using the original library when using Clover. No source code of the original library was modified.

==================================================================================
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
File: Core.js

Description:

Provides common utility functions and the Class object used internally by the library.

Also provides the <TreeUtil> object for manipulating JSON tree structures


Author:

Nicolas Garcia Belmonte

Copyright:

Copyright 2008-2009 by Nicolas Garcia Belmonte.

Homepage:

<http://thejit.org>

Version:

1.1.2

License:

BSD License

> Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
>   * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
>   * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
> * Neither the name of the organization nor the
>    names of its contributors may be used to endorse or promote products
>    derived from this software without specific prior written permission.
>
> THIS SOFTWARE IS PROVIDED BY Nicolas Garcia Belmonte ``AS IS'' AND ANY
> EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
> WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
> DISCLAIMED. IN NO EVENT SHALL Nicolas Garcia Belmonte BE LIABLE FOR ANY
> DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
> (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
> LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
> ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
> (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
> SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
>
*/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1806 to-regex-range 2.1.1 2.1.1

1.1806.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1807 process-on-spawn 1.0.0
1.1807.1 Available under license:

MIT License

Copyright (c) 2019 CFWare, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1808 generate-object-property 1.2.0 1.2.0
1.1808.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
1.1809 type-detect 4.0.8
1.1809.1 Available under license :

Copyright (c) 2013 Jake Luer <jake@alogicalparadox.com> (http://alogicalparadox.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1810 component-emitter 1.1.2 1.1.2
1.1810.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
1.1811 crypto-js 4.0.0 4.0.0
1.1811.1 Available under license :

# License

[The MIT License (MIT)][MIT]

Copyright (c) 2009-2013 Jeff Mott
Copyright (c) 2013-2016 Evan Vosberg

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1812 de.huxhorn.lilith.data.converter 0.9.44
1.1812.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort
to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would
otherwise be required to provide such information under section 6
of the GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the Combined
Work produced by recombining or relinking the Application with a
modified version of the Linked Version. (If you use option 4d0, the
Installation Information must accompany the Minimal Corresponding
Source and Corresponding Application Code. If you use option 4d1,
you must provide the Installation Information in the manner
specified by section 6 of the GNU GPL for conveying Corresponding
Source.)


You may place library facilities that are a work based on the
Library side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library facilities,
conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the
accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the
Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

/*
 * Lilith - a log event viewer.
 * Copyright (C) 2007-2011 Joern Huxhorn
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU Lesser General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/*
 * Copyright 2007-2011 Joern Huxhorn
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
1.1813 slash 1.0.0 1.0.0

1.1813.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1814 nice-try 1.0.5 1.0.5

1.1814.1 Available under license:

The MIT License (MIT)

Copyright (c) 2018 Tobias Reich

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1815 lodash.get 4.4.2 4.4.2

1.1815.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample
code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.1816 serve-static 1.14.1 1.14.1

1.1816.1 Available under license:

(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1817 drill-storage-hbase 1.19.0
1.1817.1 Available under license:

Drill : Contrib : Storage : HBase
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
   distribution, then any Derivative Works that You distribute must
   include a readable copy of the attribution notices contained
   within such NOTICE file, excluding those notices that do not
   pertain to any part of the Derivative Works, in at least one
   of the following places: within a NOTICE text file distributed
   as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1818 color-convert 0.5.3 0.5.3
1.1818.1 Available under license :
Copyright (c) 2011 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1819 jackson-dataformat-cbor 2.6.7
1.1819.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.1820 grpc-netty-shaded 1.27.1

1.1820.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1821 parse-color 1.0.0 1.0.0

1.1821.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1822 escodegen 1.14.3 1.14.3
1.1822.1 Available under license:

Copyright (C) 2012 Yusuke Suzuki (twitter: @Constellation) and other contributors.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1823 estraverse 4.2.0

1.1823.1 Available under license:
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1824 is-data-descriptor 0.1.4 0.1.4

1.1824.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.1825 pre-push 0.1.1 0.1.1

1.1825.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Arnout Kazemier, Martijn Swaagman, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1826 mchange-commons-java 0.2.15

1.1826.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
   Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.
Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must
be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under
3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the ”work that uses the Library” must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.1827 javaewah 0.3.2
1.1827.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2009-2012, Daniel Lemire
 * Licensed under APL 2.0.
 */

Found in path(s):
* /opt/cola/permits/1258286299_1643052479.7/0/javaewah-0-3-2-sources-jar/javaewah/BufferedRunningLengthWord.java
* /opt/cola/permits/1258286299_1643052479.7/0/javaewah-0-3-2-sources-jar/javaewah/RunningLengthWord.java
* /opt/cola/permits/1258286299_1643052479.7/0/javaewah-0-3-2-sources-jar/javaewah/EWAHIterator.java
* /opt/cola/permits/1258286299_1643052479.7/0/javaewah-0-3-2-sources-jar/javaewah/EWAHCompressedBitmap.java
* /opt/cola/permits/1258286299_1643052479.7/0/javaewah-0-3-2-sources-jar/javaewah/IntIterator.java

1.1828 jsreport-child-templates 1.4.0 1.4.0
1.1828.1 Available under license:
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified
version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:
0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1829 fs-readdir-recursive 1.1.0
1.1829.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1830 babel-helper-remap-async-to-generator 6.24.1

1.1830.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The "Software" is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1831 external-editor 3.1.0 3.1.0

1.1831.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Kevin Gravier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1832 y18n 5.0.8 5.0.8

1.1832.1 Available under license:

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

1.1833 x-xss-protection 1.0.0 1.0.0
1.1833.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2015 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1834 aspect-j 1.8.13
1.1834.1 Available under license :

/* **************************************************************************
 * Copyright (c) 2005 Contributors.
 * All rights reserved.
 * This program and the accompanying materials are made available
 * under the terms of the Eclipse Public License v1.0
 * which accompanies this distribution and is available at
 *
 * Contributors:
 * Adrian ColyerInitial implementation
 */
package org.aspectj.internal.lang.annotation;
Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering,
such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor
("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims,
lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by
the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a
commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any
actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a)
promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to
control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The
Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That
Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or
offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's
responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay
any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS
IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED
INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-
INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is
solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks
associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of
program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and
unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS),
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED
HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or
enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes
such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such
litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.1835 figures 2.0.0 2.0.0
1.1835.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1836 convert-source-map 1.5.1 1.5.1

1.1836.1 Available under license:

Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1837 metrics 2.2.0

1.1837.1 Available under license:

Apache HBase - Metrics Implementation
Copyright 2007-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby
assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:
a) it must be made available under this Agreement, or if the
Program (i) is combined with other material in a separate file or
files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in
Exhibit A of this Agreement, then the Program may be made available
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product
offering should do so in a manner which does not create potential
liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program
in a commercial product offering. The obligations in this section do not
apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified
Contributor must: a) promptly notify the Commercial Contributor in
writing of such claim, and b) allow the Commercial Contributor to control,
and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may
participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial
product offering, Product X. That Contributor is then a Commercial
Contributor. If that Commercial Contributor then makes performance
claims, or offers warranties related to Product X, those performance
claims and warranties are such Commercial Contributor's responsibility
alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance
claims and warranties, and if a court requires any other Contributor to
pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's
rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part
thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0
## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.1839 parquet-hadoop-bundle 1.10.0

**1.1839.1 Available under license:**

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the Appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product depends on Apache Thrift and includes it in this binary artifact.

Copyright: 2006-2010 The Apache Software Foundation.
Home page: https://thrift.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

This product depends on SLF4J and includes SLF4J in this binary artifact. SLF4J is a simple logging facade for Java.

Copyright: 2004-2013 QOS.ch.
Home page: http://www.slf4j.org/
License: http://slf4j.org/license.html (MIT license)

The following is the SLF4J license (MIT):

Copyright (c) 2004-2013 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This project includes code from Daniel Lemire's JavaFastPFOR project in this binary artifact. The "Lemire" bit packing classes produced by parquet-generator are derived from the JavaFastPFOR project.

Copyright: 2013 Daniel Lemire
Home page: http://lemire.me/en/
Project page: https://github.com/lemire/JavaFastPFOR
License: Apache License Version 2.0 http://www.apache.org/licenses/LICENSE-2.0

This product depends on fastutil and includes it in this binary artifact.
Fastutil provides type-specific collection implementations.

Copyright: 2002-2014 Sebastiano Vigna
Home page: http://fastutil.di.unimi.it/
License: http://www.apache.org/licenses/LICENSE-2.0.html

This product depends on Jackson and includes it in this binary artifact.
Jackson is a high-performance JSON processor.

Copyright: 2007-2015 Tatu Saloranta and other contributors
Home page: http://jackson.codehaus.org/
License: http://www.apache.org/licenses/LICENSE-2.0.txt

1.1840 @babel/plugin-transform-regenerator
7.16.0
1.1840.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1841 which-boxed-primitive 1.0.2

1.1841.1 Available under license:

MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1842 ws 1.1.5 1.1.5

1.1842.1 Available under license :

## License

(The MIT License)

Copyright (c) 2011 Einar Otto Stangvik &lt;einaros@gmail.com&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1843 babel-plugin-syntax-class-properties

6.13.0

1.1843.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The <<Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1844 domelementtype 1.3.0 1.3.0

1.1844.1 Available under license :

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1845 immutability-helper 2.7.1 2.7.1

1.1845.1 Available under license:
MIT License

Copyright (c) 2017 Moshe Kolodny

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1846 htrace-core4 4.2.0-incubating

1.1846.1 Available under license:

htrace-core4
Copyright 2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution.”

"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]"
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1847 semver 6.3.0 6.3.0
1.1847.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1848 aws-java-sdk-dynamodb 1.11.375
1.1848.1 Available under license :
No license file was found, but licenses were detected in source scan.

/
* Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
* http://aws.amazon.com/apache2.0
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.

* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/BetweenCondition.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/ArrayIndexElement.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/B.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/AndCondition.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/AddAction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/xspec/package-info.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/document/internal/IndexScanImpl.java

No license file was found, but licenses were detected in source scan.

/*/ 

* Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.

* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDB.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/datamodeling/AbstractEnumMarshaller.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBMapper.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/AbstractDynamoDBMapper.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/IDynamoDBMapper.java

No license file was found, but licenses were detected in source scan.

ATEGORY 2010-2018 Amazon.com, Inc. or its affiliates. All Rights
* Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/BatchLoadContext.java

No license file was found, but licenses were detected in source scan.

ATEGORY 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/AmazonDynamoDBClientConfigurationFactory.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBAutoGenerateStrategy.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
* 
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConvertedTimestamp.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBAutoGeneratedDefault.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTyped.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/StandardTypeConverters.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/StandardBeanProperties.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBKeyed.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBFlattened.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBMapperTableModel.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/StandardModelFactories.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConvertedJson.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/UpdateGlobalTableResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConvertedEnum.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConverterFactory.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConverter.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConverterFactory.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/datamodeling/ConvertibleType.java

Open Source Used In DNAC Platform Arica 8778
jar/com/amazonaws/services/dynamodbv2/model/ScalarAttributeType.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ReplicaGlobalSecondaryIndexSettingsDescriptionJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateGlobalTableRequestProtocolMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/GlobalTableMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/ReplicaStatus.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateTimeToLiveResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/BackupStatus.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ListTagsOfResourceRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/BackupStatus.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateBackupRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateTableRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/AutoScalingTargetTrackingScalingPolicyConfigurationUpdateMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateBackupRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/DescribeLimitsRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateTableRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/DescribeGlobalTableResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateBackupRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/CreateTableRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ReplicaSettingsDescription.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/DescribeGlobalTableResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/RestoreTableFromBackupRequestProtocolMarshaller.java

Open Source Used In DNAC Platform Arica 8780
Open Source Used In DNAC Platform Arica 8789

* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/AbstractAmazonDynamoDBAsync.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/ListStreamsResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/Waiters/TableExists.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/KeySchemaElementJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/SSESpecification.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/GetItemRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/DescribeTimeToLiveResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/GlobalSecondaryIndexDescription.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/DescribeGlobalTableRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/SSERequirement.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/ProvisionedThroughput.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/ListGlobalTablesRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/ItemCollectionMetricsMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/BackupDescriptionMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/ProvisionedThroughput.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/UpdateItemResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/ItemCollectionMetricsMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/ListGlobalTablesRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/KeysAndAttributesMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/ProvisionedThroughput.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateTableRequestProtocolMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/ProvisionedThroughput.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1.jar/com/amazonaws/services/dynamodbv2/model/transform/ProvisionedThroughput.java
jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateGlobalSecondaryIndexActionJsonUnmarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/UpdateGlobalTableRequest.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/DescribeStreamRequest.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateGlobalTableSettingsRequestMarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/GlobalTableGlobalSecondaryIndexSettingsUpdate.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateGlobalTableSettingsRequestProtocolMarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ProjectionMarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/GlobalSecondaryIndexUpdateMarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/TableDescriptionMarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ProjectionType.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/ReplicaGlobalSecondaryIndexSettingsDescription.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/transform/ReplicaGlobalSecondaryIndexSettingsUpdateJsonUnmarshaller.java
  * /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/model/TimeToLiveDescription.java
jar/com/amazonaws/services/dynamodbv2/model/BatchGetItemResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/ProvisionedThroughputJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/StreamJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/ShardMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/ReplicaSettingsDescriptionMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/DescribeStreamResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/GlobalSecondaryIndexJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/AutoScalingPolicyDescriptionMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/ListBackupsRequestProtocolMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/TableNotFoundException.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/CreateGlobalSecondaryIndexAction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/DeleteRequestMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/BackupDetailsMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/TableDefinition.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/ListTablesRequestProtocolMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/transform/AttributeDefinitionMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/TableUpdate.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/AutoScalingSettingsUpdate.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/TableUpdateRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
  jar/com/amazonaws/services/dynamodbv2/model/TableUpdateRequest.java

jar/com/amazonaws/services/dynamodbv2/model/transform/ConsumedCapacityMarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/OperationType.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateTableResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/ResourceNotFoundException.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/transform/ListStreamsResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/DeleteItemRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/transform/RestoreTableFromBackupResultJsonUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/RestoreTableToPointInTimeResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/TableAlreadyExistsException.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/CreateGlobalTableResult.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/BatchGetItemRequest.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/AmazonDynamoDBAsyncClientBuilder.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/model/transform/UpdateItemRequestMarshaller.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBTypeConvertedEpochDate.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/DynamoDBQueryExpression.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/AttributeTransformer.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/QueryResultPage.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/ScanResultPage.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/datamodeling/AttributeTransformerChain.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-
jar/com/amazonaws/services/dynamodbv2/document/ItemUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2018 Amazon Technologies, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1- jar/com/amazonaws/services/dynamodbv2/metrics/DynamoDBRequestMetric.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1- jar/com/amazonaws/services/dynamodbv2/util/TableUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1- jar/com/amazonaws/services/dynamodbv2/datamodeling/unmarshallers/ShortSetUnmarshaller.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1- jar/com/amazonaws/services/dynamodbv2/datamodeling/marshallers/NumberSetToNumberSetMarshaller.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon Technologies, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/OrCondition.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/NegationCondition.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/MinusOperation.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/NamedElement.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/PathElement.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/RemoveAction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/IfNotExistsFunction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/S.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/RemoveAction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/IfNotExistsFunction.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/PathOperand.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/BinaryOperation.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/BOOL.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/DeleteItemExpressionSpec.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/PutItemExpressionSpec.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/PathOperand.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/BinaryOperation.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/NS.java
* /opt/cola/permits/1180370909_1626286006.85/0/aws-java-sdk-dynamodb-1-11-375-sources-1-jar/com/amazonaws/services/dynamodbv2/xspec/SS.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
1.1849 jansi-osx 1.8
1.1849.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2017 the original author(s).
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257860178_1643019491.32/0/jansi-osx-1-8-sources-jar/org/fusesource/jansi/internal/Kernel32.java
* /opt/cola/permits/1257860178_1643019491.32/0/jansi-osx-1-8-sources-jar/org/fusesource/jansi/internal/WindowsSupport.java
* /opt/cola/permits/1257860178_1643019491.32/0/jansi-osx-1-8-sources-jar/org/fusesource/jansi/internal/CLibrary.java
1.1850 archiver-utils 1.3.0 1.3.0

1.1850.1 Available under license:

Copyright (c) 2015 Chris Talkington.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1851 process-nextick-args 2.0.1 2.0.1

1.1851.1 Available under license:

# Copyright (c) 2015 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1852 apib2swagger 1.12.0 1.12.0

1.1852.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Keisuke Minami

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1853 is-core-module 2.0.0 2.0.0

1.1853.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Dave Justice

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1854 libxml 2.9.3+dfsg1 1ubuntu0.7

1.1854.1 Available under license:
This package was debianized by Vincent Renardias <vincent@waw.com> on Sat, 26 Sep 1998 16:50:54 +0200

It was downloaded from ftp://xmlsoft.org/libxml2/

==========================================

Except where otherwise noted in the source code (trio files, hash.c and list.c) covered by a similar licence but with different Copyright notices:

Copyright (C) 1998-2002 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

trio.c, trio.h:
Copyright (C) 1998 Bjorn Reese and Daniel Stenberg.

Permission to use, copy, modify, and distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED `AS IS' AND WITHOUT ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND
CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

==========================

trio.h:
Copyright (C) 2000 Bjorn Reese and Daniel Stenberg.

Permission to use, copy, modify, and distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED `AS IS' AND WITHOUT ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND
CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

==========================

hash.c:
Copyright (C) 2000 Bjorn Reese and Daniel Veillard.

Permission to use, copy, modify, and distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED `AS IS' AND WITHOUT ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND
CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

==========================

list.c:
Copyright (C) 2000 Gary Pennington and Daniel Veillard.

Permission to use, copy, modify, and distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THIS SOFTWARE IS PROVIDED `AS IS' AND WITHOUT ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS AND
CONTRIBUTORS ACCEPT NO RESPONSIBILITY IN ANY CONCEIVABLE MANNER.

==========================

triodef.h, trionan.c, trionan.h:
Copyright (C) 2001 Bjorn Reese <breese@users.sourceforge.net>

Permission to use, copy, modify, and distribute this software for any
1.1855 ajv 6.10.0

1.1855.1 Available under license :  
The MIT License (MIT)

Copyright (c) 2015-2017 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Open Source Used In DNAC Platform Arica

1.1856 awaitility 3.0.0

1.1856.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
1.1857 @babel/plugin-transform-spread
7.16.0
1.1857.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1858 batik-codec 1.13

1.1858.1 Available under license:

No license file was found, but licenses were detected in source scan.

# -----------------------------------------------------------------------------
#
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# This file registers PNG and JPEG codecs.
#
# $Id: org.apache.batik.ext.awt.image.spi.RegistryEntry 1790950 2017-04-11 10:54:26Z ssteiner $
# -----------------------------------------------------------------------------

org.apache.batik.ext.awt.image.codec.png.PNGRegistryEntry

org.apache.batik.ext.awt.image.codec.imageio.ImageIOPNGRegistryEntry

org.apache.batik.ext.awt.image.codec.imageio.ImageIOTIFFRegistryEntry

Org in path(s):
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/META-INF/services/org.apache.batik.ext.awt.image.spi.RegistryEntry

No license file was found, but licenses were detected in source scan.

# -----------------------------------------------------------------------------
#
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# This file registers PNG and JPEG writer classes.
#
# $Id: org.apache.batik.ext.awt.image.spi.ImageWriter 1733416 2016-03-03 07:07:13Z gadams $
#
# ---

# NOTE: the "codec" package is deprecated, there entries are kept here for compatibility with older JVM versions
# (uses "sun.image", which is only supported in Sun Java implementations and was retired in JDK 7)
#org.apache.batik.ext.awt.image.codec.png.PNGImageWriter
org.apache.batik.ext.awt.image.codec.imageio.ImageIOPNGImageWriter
org.apache.batik.ext.awt.image.codec.imageio.ImageIOTIFFImageWriter
org.apache.batik.ext.awt.image.codec.imageio.ImageIOJPEGImageWriter

Found in path(s):
No license file was found, but licenses were detected in source scan.

/

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/png/PNGImageDecoder.java
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/imageio/ImageIOTIFFRegistryEntry.java
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/png/PNGSuggestedPaletteEntry.java
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/util/ForwardSeekableStream.java
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/util/PropertyUtil.java
* /opt/cola/permits/1146878584_1616835244.27/0/batik-codec-1-13-sources-1-jar/org/apache/batik/ext/awt/image/codec/imageio/ImageIOJPEGRegistryEntry.java

No license file was found, but licenses were detected in source scan.

1.1859 parquet-column 1.12.0
1.1859.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product depends on fastutil and includes it in this binary artifact.
Fastutil provides type-specific collection implementations.

Copyright: 2002-2014 Sebastiano Vigna
Home page: http://fasutil.di.unimi.it/
License: http://www.apache.org/licenses/LICENSE-2.0.html

1.1860 influx 5.6.3 5.6.3
1.1860.1 Available under license :
MIT License

Copyright (c) 2016 Connor Peet

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1861 batik-squiggle 1.13
1.1861.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are
under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of
fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source
code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form,
including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1862 lodash-foreach 4.5.0 4.5.0

1.1862.1 Available under license:

Copyright jQuery Foundation and other contributors &lt;https://jquery.org/&gt;

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors &lt;http://underscorejs.org/&gt;

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

**1.1863 globals 9.18.0 9.18.0**

**1.1863.1 Available under license:**

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1864 istack-commons 3.0.7

1.1864.1 Available under license:

Found license 'General Public License 2.0' in 'Copyright (c) 1997-2018 Oracle and/or its affiliates. All rights
reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only
("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may
not use this file except in compliance with the License. You can Oracle designates this particular file as subject to
the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2018 Oracle
and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General
Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL")
(collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle
designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2
section of the License'

1.1865 ecc-jsbn 0.1.2 0.1.2

1.1865.1 Available under license:

Licensing
---------

This software is covered under the following copyright:

/*
 * Copyright (c) 2003-2005 Tom Wu
 * All Rights Reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY
 * WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
 *
 * IN NO EVENT SHALL TOM WU BE LIABLE FOR ANY SPECIAL, INCIDENTAL,
 * INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER
 * RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF
* THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* In addition, the following condition applies:
*
* All redistributions must retain an intact copy of this copyright notice
* and disclaimer.
*/

Address all questions regarding this license to:

Tom Wu
tjw@cs.Stanford.EDU
The MIT License (MIT)

Copyright (c) 2014 Jeremie Miller

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1866 maxmind-db 1.4.0

1.1866.1 Available under license:
No license file was found, but licenses were detected in source scan.

<name>Apache License 2.0</name>
<url>http://www.apache.org/licenses/LICENSE-2.0.html</url>

Found in path(s):
* /opt/cola/permits/1140689161_1614870595.69/0/maxmind-db-1-4-0-sources-1-jar/META-INF/maven/com.maxmind.db/maxmind-db/pom.xml
1.1867 hbase-metrics-api 2.4.2
1.1867.1 Available under license:

Apache HBase - Metrics API
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1868 jmespath-java 1.11.837

1.1868.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permitis/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathSubExpression.java
* /opt/cola/permitis/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathLiteral.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/OpGreaterThanOrEqualTo.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/OpNotEquals.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/Comparator.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/ObjectMapperSingleton.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/OpLessThan.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/NumericComparator.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathField.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/InvalidTypeException.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathContainsFunction.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathFlatten.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathFunction.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/CamelCaseUtils.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathAndExpression.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathValueProjection.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathExpression.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/OpLessThanOrEqualTo.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JSesPathMultiSelectList.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathProjection.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathVisitor.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathLengthFunction.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathNotExpression.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathIdentity.java
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/com/amazonaws/jmespath/JmesPathFilter.java

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):
* /opt/cola/permits/1137760046_1614121472.51/0/jmespath-java-1-11-837-sources-jar/META-INF/maven/com.amazonaws/jmespath-java/pom.xml

---

1.1869 domutils 1.5.1 1.5.1

1.1869.1 Available under license :

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

1.1870 babel-runtime 6.26.0 6.26.0

1.1870.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1871 okio 1.13.0

1.1871.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1000000590_1646171445.35/0/okio-1-13-0-sources-2-jar/okio/Utf8.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 Square, Inc.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):

*/

Copyright (C) 2014 Square, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1000000590_1646171445.35/0/okio-1-13-0-sources-2-jar/okio/ByteString.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1000000590_1646171445.35/0/okio-1-13-0-sources-2-jar/okio/Base64.java

1.1872 babel-plugin-transform-es2015-parameters 6.24.1

1.1872.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrite noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1873 tpch-sample-data 1.19.0
1.1873.1 Available under license :

Drill : Contrib : Data : TPCH Sample
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1874.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1875 ms 0.7.2 0.7.2
1.1875.1 Available under license :
The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1876 color-convert 1.9.2

1.1876.1 Available under license:

Copyright (c) 2011-2016 Heather Arthur <fayearthur@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1877 emoji-regex 7.0.3 7.0.3

1.1877.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1878 globals 11.7.0
1.1878.1 Available under license:
MIT License

Copyright (c) Sindresorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1879 object.values 1.1.5
1.1879.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1880 source-map-support 0.4.18 0.4.18

1.1880.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Evan Wallace

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1881 generate-function 2.3.1

1.1881.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1882 asap 2.0.6 2.0.6
1.1882.1 Available under license :

Copyright 20092014 Contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1883 drill-format-excel 1.19.0
1.1883.1 Available under license :

Drill : Contrib : Format : Excel
Copyright 2021 The Apache Software Foundation
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

Open Source Used In DNAC Platform Arica 8856
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Open Source Used In DNAC Platform Arica 8858
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.1884 prepend-http 2.0.0

1.1884.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1885 commons-compiler 3.0.11

1.1886 p-finally 1.0.0 1.0.0

1.1886.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1887 copy-descriptor 0.1.1 0.1.1
1.1887.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1888 signal-exit 3.0.3 3.0.3
1.1888.1 Available under license:
The ISC License

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1889 parse-filepath 1.0.2 1.0.2
1.1889.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1890 os-tmpdir 1.0.2 1.0.2

1.1890.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1891 split-on-first 1.1.0 1.1.0

1.1891.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1892 regjsgen 0.2.0

1.1892.1 Available under license:

Copyright 2014 Benjamin Tan <demoneaux@gmail.com> (https://d10.github.io/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1893 netty-tcnative-boringssl-static

2.0.26.Final

1.1893.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources.jar/io/netty/internal/tcnative/SniHostNameMatcher.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources.jar/io/netty/internal/tcnative/NativeStaticallyReferencedJniMethods.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources.jar/io/netty/internal/tcnative/CertificateCallback.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources.jar/io/netty/internal/tcnative/CertificateCallback.java
No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/SSLPrivateKeyMethodTask.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/SSLTask.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/CertificateCallbackTask.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/CertificateVerifierTask.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/SSLPrivateKeyMethodDecryptTask.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tcnative/SSLPrivateKeyMethodSignTask.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
BoringSSL-Revision: 6f3e034fe58ce3bca3c1c32be6d603a6206c89af
Bundle-Description: A Mavenized fork of Tomcat Native which incorporates various patches. This artifact is statically linked to BoringSSL and Apache APR.
Apr-Version: 1.6.5
Automatic-Module-Name: io.netty.tcnative.boringssl
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0
Bundle-SymbolicName: io.netty.tcnative-boringssl-static
Archiver-Version: Plexus Archiver
Built-By: norman
Bnd-LastModified: 1569053001934
Bundle-ManifestVersion: 2
Bundle-Name: Netty/TomcatNative [BoringSSL - Static]
Bundle-Version: 2.0.26.Final
BoringSSL-Branch: chromium-stable
Bundle-NativeCode: META-INF/native/libnetty_tcnative.jnilib;osname=osx;processor=x86_64
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_212

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tnative-boringssl-static-2-0-26-final-sources-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

/

* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tnative/Library.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tnative/Buffer.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tnative/SSLContext.java
* /opt/cola/permits/1258272134_1643050031.94/0/netty-tcnative-boringssl-static-2-0-26-final-sources-
  jar/io/netty/internal/tnative/SSL.java

1.1894 cookiejar 2.1.2 2.1.2

1.1894.1 Available under license:
The MIT License (MIT)
Copyright (c) 2013 Bradley Meck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1895 event-stream 3.3.4 3.3.4
1.1895.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1896 jnr-x86asm 1.0.2
1.1897 kafka-clients 2.8.0

1.1897.1 Available under license:

Apache Kafka
Copyright 2021 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted by them:
Copyright (C) 2018 Lightbend Inc. <https://www.lightbend.com>
Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:
clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java
Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1898 drill-hive-exec-shaded 1.19.0

1.1898.1 Available under license:

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Drill : Contrib : Storage : Hive : Exec Shaded
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0
To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

Apache Commons Codec
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

1.1899 babel-plugin-transform-es2015-
shorthand-properties 6.24.1
Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA

CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.1900 node-signpdf 1.3.2 1.3.2

1.1900.1 Available under license:

MIT LICENSE
Copyright (c) 2014 Devon Govett

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License
Copyright (c) 2019 The Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1901 flat 4.1.1

1.1901.1 Available under license:
Copyright (c) 2014, Hugh Kennedy
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1902 commons-pool 1.6

1.1902.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Pool
Copyright 2001-2012 The Apache Software Foundation

This product includes software developed by
1.1903 passport-http 0.3.0 0.3.0

1.1903.1 Available under license:
(The MIT License)

Copyright (c) 2011-2013 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1904 detect-indent 4.0.0 4.0.0

1.1904.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1905 metrics 4.0.2

1.1906 natty 0.13
1.1906.1 Available under license:
No license file was found, but licenses were detected in source scan.

<name>MIT License</name>

Found in path(s):
* /opt/cola/permits/1136702688_1613864332.59/0/natty-0-13-sources-3-jar/META-INF/maven/com.joestelmach/natty/pom.xml

1.1907 ansistyles 0.1.3 0.1.3
1.1907.1 Available under license:
Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1908 invert-kv 1.0.0 1.0.0

1.1908.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1909 hadoop-mapreduce-client-core 3.2.2

1.1909.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/awslabs/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4,
which has the following notices:

    The Netty Project
    =================

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the
components that this product depends on.

------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has
been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain
Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* HOMEPAGE:
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSEprotobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:
This product contains a modified portion of ‘JDOM 1.1’, which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons(codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright:Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
  Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.
The binary distribution of this product bundles binaries of
  Jetty :: Http Utility 9.3.19.,
  Jetty :: IO Utility 9.3.19.,
  Jetty :: Security 9.3.19.,
  Jetty :: Server Core 9.3.19.,
  Jetty :: Servlet Handling 9.3.19.,
  Jetty :: Utilities 9.3.19.,
  Jetty :: Utilities :: Ajax,
  Jetty :: Webapp Application Support 9.3.19.,
  Jetty :: XML utilities 9.3.19.,
which has the following notices:

  * ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

  * The Apache 2.0 License
    http://www.apache.org/licenses/LICENSE-2.0.html
    
    and

  * The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
  * org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
  * org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
  * org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle
The following artifacts are CDDL + GPLv2 with classpath exception. 
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions. 
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.
org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUntime Library Exception"
== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of
Xerces2 Java Parser 2.9.1,
which has the following notices:

=================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License, ==
==  Version 2.0, in this case for the Apache Xerces Java distribution. ==
=================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
  - voluntary contributions made by Paul Eng on behalf of the
    Apache Software Foundation that were originally developed at iClick, Inc.,
    software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
  * Logback: the reliable, generic, fast and flexible logging framework.
    Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
  * Apache HBase
    Copyright 2007-2015 The Apache Software Foundation
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:

--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 2008-2015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 2007-2008 Robert Niederreiter, Jens Klein
# Copyright 2004-2005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of Plexus Cipher: encryption/decryption Component 1.4, which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of software.amazon.ion:ion-java 1.0.1, which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of joda-time:joda-time:2.9.9 which has the following notices:
* =-------------------------------------------------------------------------------------------------------------------------------------=
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  =-------------------------------------------------------------------------------------------------------------------------------------=

This product includes software developed by Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64,
LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/*++
 *
 * Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
 *
 * This software program is licensed subject to the BSD License,
 * available at http://www.opensource.org/licenses/bsd-license.html
 *
 * Abstract: The main routine
 *
 *--*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/{lz4.h,lz4.c,lz4hc.h,lz4hc.c},

/*
 Lz4 - Fast LZ compression algorithm
 Header File
 Copyright (C) 2011-2014, Yann Collet.
 BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

 * Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following disclaimer
   in the documentation and/or other materials provided with the
distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc., nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

---------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its contributors
  may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------------------------------------------------------------

Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni
(https://github.com/fusesource/leveldbjni), which is available under the
following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of FuseSource Corp. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS `AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

==============

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the
Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
  hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
  hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js
--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
datatables v1.10.8
em-helpers v0.5.13
em-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.13.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
moment.js v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

-------------------------------

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains jQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1,jre7

-------------------------------

MIT License

Copyright (c) 2003-2017 Optimatika
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0

--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:

hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

--------------------------------------------------------------------------------

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the
Hypersonic SQL Group.”

The binary distribution of this product bundles these dependencies under the
following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

--------------------------------------------------------------------------------
(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0
1.Definitions.

1.1.Contributormeans each individual or entity
that creates or contributes to the creation of
Modifications.

1.2.Contributor Versionmeans the combination of the
Original Software, prior Modifications used by a Contributor (if any), and the
Modifications made by that particular Contributor.

1.3.Covered
Software means (a) the Original Software, or (b) Modifications, or (c) the
combination of files containing Original Software with files containing
Modifications, in each case including portions
thereof.

Open Source Used In DNAC Platform Arica 8919
1.4. **Executable** means the Covered Software in any form other than Source Code.

1.5. **Initial Developer** means the individual or entity that first makes Original Software available under this License.

1.6. **Larger Work** means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. **License** means this document.

1.8. **Licensable** means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. **Modifications** means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. **Original Software** means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. **Patent Claims** means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. **Source Code** means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. **You** (or **Your**) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.
2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer.
and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the
license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE,
SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its
utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the
following:
A. Any file that results from an addition to, deletion from or modification of
the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous
Modification; or
C. Any new file that is contributed or otherwise made available under the terms
of this License.

1.10. Original Software means the Source Code and Executable form of
computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter
acquired, including without limitation, method, process, and apparatus claims,
in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in
which modifications are made and (b) associated documentation included in or
with such code.

1.13. You (or Your) means an individual or a legal entity exercising
rights under, and complying with all of the terms of, this License. For legal
entities, You includes any entity which controls, is controlled by, or is
under common control with You. For purposes of this definition, control
means (a) the power, direct or indirect, to cause the direction or management of
such entity, whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial ownership of such
entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to
third party intellectual property claims, the Initial Developer hereby grants
You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual
property rights (other than patent or trademark) Licensable by Initial
Developer, to use, reproduce, modify, display, perform, sublicense and
distribute the Original Software (or portions thereof), with or without
Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims
infringed by the making, using or selling of Original Software, to make, have
made, use, practice, sell, and offer for sale, and/or otherwise dispose of the
Original Software (or portions thereof).
(c) The licenses granted in Sections
2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms
of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or (2) for
infringements caused by: (i) the modification of the Original Software, or (ii)
the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted:

(1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the
Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under
which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within
such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning
subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:

Protocol Buffers API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

**THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS**
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04
------------------------------------------------------------------------------------------------------------------------

By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation,
with or without modification, for any purpose and without fee or royalty is
hereby granted, provided that you include the following on ALL copies of the
software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the
  redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and
  conditions. If none exist, the W3C Software Short Notice should be included
  (hypertext is preferred, text is permitted) within the body of any redistributed
  or derivative code.
- Notice of any changes or modifications to the files, including the date changes
  were made. (We recommend you provide URIs to the location from which the code is
  derived.)

The binary distribution of this product bundles these dependencies under the
following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.
4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:

JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast,
transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange),
Royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats.

All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g. "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.
5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS
THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND
CONCERNING
THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT
LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY,
OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME
JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH
EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN
NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL,
INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS
LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon
any breach by You of the terms of this License. Individuals or entities who have
received Derivative Works or Collective Works from You under this License,
however, will not have their licenses terminated provided such individuals or
entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7,
and 8 will survive any termination of this License.
Subject to the above terms and conditions, the license granted here is perpetual
(for the duration of the applicable copyright in the Work). Notwithstanding the
above, Licensor reserves the right to release the Work under different license
terms or to stop distributing the Work at any time; provided, however that any
such election will not serve to withdraw this License (or any other license that
has been, or is required to be, granted under the terms of this License), and
this License will continue in full force and effect unless terminated as stated
above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective
Work, the Licensor offers to the recipient a license to the Work on the same
terms and conditions as the license granted to You under this License.
Each time You distribute or publicly digitally perform a Derivative Work,
Licensor offers to the recipient a license to the original Work on the same
terms and conditions as the license granted to You under this License.
If any provision of this License is invalid or unenforceable under applicable
law, it shall not affect the validity or enforceability of the remainder of the
terms of this License, and without further action by the parties to this
agreement, such provision shall be reformed to the minimum extent necessary to
make such provision valid and enforceable.
No term or provision of this License shall be deemed waived and no breach
consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

--------------------------------------------------------------------------------
(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

EachContributor hereby grants You a world-wide, royalty-free, non-exclusive
license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its
Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such
3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of anyContributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.
5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous
This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If you choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

--------------------------------------------------------------------------------
/*--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

/*

The binary distribution of this product bundles these dependencies under the following license:
Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions
a. "Adaptation" means a work based upon the Work, or upon the Work and
other pre-existing works, such as a translation, adaptation,
derivative work, arrangement of music or other alterations of a
literary or artistic work, or phonogram or performance and includes
cinematographic adaptations or any other form in which the Work may be
recast, transformed, or adapted including in any form recognizably
derived from the original, except that a work that constitutes a
Collection will not be considered an Adaptation for the purpose of
this License. For the avoidance of doubt, where the Work is a musical
work, performance or phonogram, the synchronization of the Work in
timed-relation with a moving image ("synching") will be considered an
Adaptation for the purpose of this License.
b. "Collection" means a collection of literary or artistic works, such as
encyclopedias and anthologies, or performances, phonograms or
broadcasts, or other works or subject matter other than works listed
in Section 1(f) below, which, by reason of the selection and
arrangement of their contents, constitute intellectual creations, in
which the Work is included in its entirety in unmodified form along
with one or more other contributions, each constituting separate and
independent works in themselves, which together are assembled into a
collective whole. A work that constitutes a Collection will not be
considered an Adaptation (as defined above) for the purposes of this
License.
c. "Distribute" means to make available to the public the original and
copies of the Work or Adaptation, as appropriate, through sale or
other transfer of ownership.
d. "Licensor" means the individual, individuals, entity or entities that
offer(s) the Work under the terms of this License.
e. "Original Author" means, in the case of a literary or artistic work,
the individual, individuals, entity or entities who created the Work
or if no individual or entity can be identified, the publisher; and in
addition (i) in the case of a performance the actors, singers,
musicians, dancers, and other persons who act, sing, deliver, declaim,
play in, interpret or otherwise perform literary or artistic works or
expressions of folklore; (ii) in the case of a phonogram the producer
being the person or legal entity who first fixes the sounds of a
performance or other sounds; and, (iii) in the case of broadcasts, the
organization that transmits the broadcast.
f. "Work" means the literary and/or artistic work offered under the terms
of this License including without limitation any production in the
literary, scientific and artistic domain, whatever may be the mode or
form of its expression including digital form, such as a book,
pamphlet and other writing; a lecture, address, sermon or other work
of the same nature; a dramatic or dramatico-musical work; a
choreographic work or entertainment in dumb show; a musical
composition with or without words; a cinematographic work to which are
assimilated works expressed by a process analogous to cinematography;
a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a
modification could indicate "The original work has been modified.";
c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
d. to Distribute and Publicly Perform Adaptations.
e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You
must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the
applicable national law, to enable You to reasonably exercise Your
right under Section 3(b) of this License (right to make Adaptations)
but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR
OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY
KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE,
INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF
LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS,
WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION
OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE
LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR
ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES
ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS
BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate
automatically upon any breach by You of the terms of this License.
Individuals or entities who have received Adaptations or Collections
from You under this License, however, will not have their licenses
terminated provided such individuals or entities remain in full
compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will
survive any termination of this License.
b. Subject to the above terms and conditions, the license granted here is
perpetual (for the duration of the applicable copyright in the Work).
Notwithstanding the above, Licensor reserves the right to release the
Work under different license terms or to stop distributing the Work at
any time; provided, however that any such election will not serve to
withdraw this License (or any other license that has been, or is
required to be, granted under the terms of this License), and this
License will continue in full force and effect unless terminated as
stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection,
the Licensor offers to the recipient a license to the Work on the same
terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor
offers to the recipient a license to the original Work on the same
terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in
compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

**1.1910 crc 3.8.0 3.8.0**

**1.1910.1 Available under license:**

The MIT License (MIT)

Copyright 2014 Alex Gorbatchev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
1.1911 object-assign 3.0.0 3.0.0

1.1911.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1912 locate-path 5.0.0 5.0.0

1.1912.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1913 jsreport-scheduling 2.5.0 2.5.0

1.1913.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a
facility refers to a function or data to be supplied by an Application
that uses the facility (other than as an argument passed when the
facility is invoked), then you may convey a copy of the modified
version:

a) under this License, provided that you make a good faith effort to
ensure that, in the event an Application does not supply the
function or data, the facility still operates, and performs
whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of
this License applicable to that copy.


The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:
a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would otherwise
be required to provide such information under section 6 of the
GNU GPL, and only to the extent that such information is
necessary to install and execute a modified version of the
Combined Work produced by recombining or relinking the
Application with a modified version of the Linked Version. (If
you use option 4d0, the Installation Information must accompany
the Minimal Corresponding Source and Corresponding Application
Code. If you use option 4d1, you must provide the Installation
Information in the manner specified by section 6 of the GNU GPL
for conveying Corresponding Source.)


You may place library facilities that are a work based on the
Library side by side in a single library together with other library
facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your
choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

---

1.1914 @babel/highlight 7.10.4 7.10.4

1.1914.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

---

Open Source Used In DNAC Platform Arica 8963
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1915 chalk 1.1.3 1.1.3

1.1915.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1916 glib 2.48.2-0ubuntu1

1.1916.1 Available under license:
This work may be reproduced and distributed in whole or in part, in any medium, physical or electronic, so as long as this copyright notice remains intact and unchanged on all copies. Commercial redistribution is permitted and encouraged, but you may not redistribute, in whole or in part, under terms more restrictive than those under which you received it. If you redistribute a modified or
translated version of this work, you must also make the source code to
the modified or translated version available in electronic form
without charge. However, mere aggregation as part of a larger work
shall not count as a modification for this purpose.

All code examples in this work are placed into the public domain,
and may be used, modified and redistributed without restriction.

BECAUSE THIS WORK IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE WORK, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE WORK "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. SHOULD THE WORK PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE WORK AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
WORK, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.
This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDEED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.1917 jsreport-xlsx 2.5.0 2.5.0
1.1917.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1918 node-modules-regexp 1.0.0
1.1918.1 Available under license:
The MIT License (MIT)

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1919 escalade 3.1.1 3.1.1

1.1919.1 Available under license:
MIT License

Copyright (c) Luke Edwards <luke.edwards05@gmail.com> (lukeed.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1920 react-pdf 3.0.6 3.0.6

1.1920.1 Available under license:
MIT License

Copyright (c) 20172018 Wojciech Maj

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls.
See http://www.jcip.net and the Creative Commons Attribution License
(http://creativecommons.org/licenses/by/2.5)
Apache HttpComponents AsyncClient
Copyright 2010-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1922 express 4.16.3 4.16.3
1.1922.1 Available under license :
(The MIT License)

Copyright (c) 2009-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2013-2014 Roman Shtylman <shtylman+expressjs@gmail.com>
Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1923 babel-eslint 6.1.2
1.1923.1 Available under license :
Copyright (c) 2014-2016 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE 
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION 
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1924 gtk 3.18.9-1ubuntu3.3

1.1924.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA

Everyone is permitted to copy and distribute verbatim copies 
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL.  It is 
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your 
freedom to share and change it.  By contrast, the GNU General Public 
Licenses are intended to guarantee your freedom to share and change 
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some 
specially designated Free Software Foundation software, and to any 
other libraries whose authors decide to use it.  You can use it for 
your libraries, too.

When we speak of free software, we are referring to freedom, not 
price.  Our General Public Licenses are designed to make sure that you 
have the freedom to distribute copies of free software (and charge for 
this service if you wish), that you receive source code or can get it 
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.
Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means
all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.
If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

d) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the source code distributed need not include anything that is normally
distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".
A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these
For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying materials or that you have already sent this user a copy.
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
1.1925.1 Available under license:

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1926 is-my-json-valid 2.17.2 2.17.2

1.1926.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.1927 dom-serializer 0.1.1 0.1.1
1.1927.1 Available under license:

License

(The MIT License)

Copyright (c) 2014 The cheeriojs contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1928 drill-jdbc-all 1.19.0
1.1928.1 Available under license:

Drill : Exec : JDBC JAR with all dependencies
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.1929 react-lifecycles-compat 3.0.4 3.0.4

1.1929.1 Available under license:

MIT License

Copyright (c) 2013-present, Facebook, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1930 converter-jackson 2.8.1

1.1930.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor `jackson-module-jaxb-annotations` module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
1.1932.1 Available under license:
Copyright (c) 2014 Hage Yaapa <http://www.hacksparrow.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1933.1 Available under license:

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>ICU License - ICU 1.8.1 and later</title>
</head>
<body bgcolor="#ffffff">
<h2>ICU License - ICU 1.8.1 and later</h2>
<p>COPYRIGHT AND PERMISSION NOTICE</p>
<p>Copyright (c) 1995-2015 International Business Machines Corporation and others</p>
<p>All rights reserved.</p>
<p>Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, publish, distribute, and/or sell
copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies
of the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.
</p>
<p>
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL
THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM,
OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE
USE OR PERFORMANCE OF THIS SOFTWARE.
</p>
<p>
Except as contained in this notice, the name of a copyright holder shall not be
used in advertising or otherwise to promote the sale, use or other dealings in
this Software without prior written authorization of the copyright holder.
</p>

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

<h2>Third-Party Software Licenses</h2>
This section contains third-party software notices and/or additional terms for licensed
third-party software components included within ICU libraries.

<h3>1. Unicode Data Files and Software</h3>

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2015 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that
(a) this copyright and permission notice appear with all copies of the Data Files or Software,
(b) this copyright and permission notice appear in associated documentation, and
(c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

<h3>2. Chinese/Japanese Word Break Dictionary Data (cjdict.txt)</h3>
<pre>
# The Google Chrome software developed by Google is licensed under the BSD license. Other software included in this distribution is provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
# Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
# Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
#
#</pre>
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
# The word list in cjdict.txt are generated by combining three word lists listed
# below with further processing for compound word breaking. The frequency is generated
# with an iterative training against Google web corpora.
#
# * Libtabe (Chinese)
#   - https://sourceforge.net/project/?group_id=1519
#   - Its license terms and conditions are shown below.
#
# * IPADIC (Japanese)
#   - http://chasen.aist-nara.ac.jp/chasen/distribution.html
#   - Its license terms and conditions are shown below.
#
#--------COPYING.libtabe ---- BEGIN-------------------
#
#/*
# Copyright (c) 1999 TaBE Project.
# Copyright (c) 1999 Pai-Hsiang Hsiao.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
#
# . Redistributions of source code must retain the above copyright
# notice, this list of conditions and the following disclaimer.
# . Redistributions in binary form must reproduce the above copyright
# notice, this list of conditions and the following disclaimer in
# the documentation and/or other materials provided with the
# distribution.
# . Neither the name of the TaBE Project nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
##

/*
 * Copyright (c) 1999 Computer Systems and Communication Lab,
 * Institute of Information Science, Academia Sinica.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * . Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 * . Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in
 *   the documentation and/or other materials provided with the
 *   distribution.
 * . Neither the name of the Computer Systems and Communication Lab
 * nor the names of its contributors may be used to endorse or
 * promote products derived from this software without specific
 * prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 */

#Copyright 1996 Chih-Hao Tsai @ Beckman Institute, University of Illinois
#c-t sai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-t sai4
#

Use, reproduction, and distribution of this software is permitted. Any copy of this software, whether in its original form or modified, must include both the above copyright notice and the following paragraphs.

Nara Institute of Science and Technology (NAIST), the copyright holders, disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness, in no event shall NAIST be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

A large portion of the dictionary entries originate from ICOT Free Software. The following conditions for ICOT Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its original form or modified, to any third party or parties, PROVIDED that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear on, or be attached to, the Program, which is distributed substantially in the same form as set out herein and that such intended distribution, if actually made, will neither violate or otherwise contravene any of the laws and regulations of the countries having jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the research and development conducted during the project and is provided to users as so produced on an experimental basis. Accordingly, the program is provided without any warranty whatsoever, whether express, implied, statutory or otherwise. The term "warranty" used herein includes, but is not limited to, any warranty of the quality, performance, merchantability and fitness for a particular purpose of the program and the nonexistence of any infringement or violation of any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.
#
Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.
#
In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.
#
--------------COPYING.ipadic-----END------------------------------------
</pre>

<h3>3. Lao Word Break Dictionary Data (laodict.txt)</h3>
<pre>
# Copyright (c) 2013 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: http://code.google.com/p/lao-dictionary/
# License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
# (copied below)
#
# This file is derived from the above dictionary, with slight modifications.
#-----------------------------------------------
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without modification,
# are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4. Burmese Word Break Dictionary Data (burmesedict.txt)

Copyright (c) 2014 International Business Machines Corporation and others. All Rights Reserved.

This list is part of a project hosted at: github.com/kanyawtech/myanmar-karen-word-lists

--------------------------------------------------------------------------------

Copyright (c) 2013, LeRoy Benjamin Sharon
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name Myanmar Karen Word Lists, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------
<h3>5. Time Zone Database</h3>
ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in section 7.

<p>7. Database Ownership</p>
The TZ database itself is not an IETF Contribution or an IETF document. Rather it is a pre-existing and regularly updated work that is in the public domain, and is intended to remain in the public domain. Therefore, BCPs 78 and 79 do not apply to the TZ Database or contributions that individuals make to it. Should any claims be made and substantiated against the TZ Database, the organization that is providing the IANA Considerations defined in this RFC, under the memorandum of understanding with the IETF, currently ICANN, may act in accordance with all competent court orders. No ownership claims will be made by ICANN or the IETF Trust on the database or the code. Any person making a contribution to the database or code waives all rights to future claims in that contribution or in the TZ Database.

</body>
</html>

## -*-makefile-*-
## BSD-specific setup (FreeBSD, OpenBSD, NetBSD, *BSD)
## Copyright (c) 1999-2013, International Business Machines Corporation and
## others. All Rights Reserved.

## Commands to generate dependency files
GEN_DEPS.c=$(CC) -E -MM $(DEFS) $(CPPFLAGS)
GEN_DEPS.cc=$(CXX) -E -MM $(DEFS) $(CPPFLAGS)
## Flags for position independent code
SHAREDLIBCFLAGS = -fPIC
SHAREDLIBCXXFLAGS = -fPIC
SHAREDLIBCPPFLAGS = -DPIC

## Additional flags when building libraries and with threads
THREADSCPPFLAGS = -D_REENTRANT
LIBCPPFLAGS =

## Compiler switch to embed a runtime search path
LD_RPATH=
LD_RPATH_PRE=-Wl,-rpath,

## Compiler switch to embed a library name
LD_SONAME = -Wl,-soname -Wl,$(notdir $(MIDDLE_SO_TARGET))

## Shared library options
LD_SOOPTIONS= -Wl,-Bsymbolic

## Shared object suffix
SO = so

## Non-shared intermediate object suffix
STATIC_O = ao

## Compilation rules
%.$(STATIC_O): $(srcdir)/%.c
$(COMPILE.c) $(STATICCPPFLAGS) $(STATICCFLAGS) -o $@ $<
%.o: $(srcdir)/%.c
$(COMPILE.c) $(DYNAMICCPPFLAGS) $(DYNAMICCFLAGS) -o $@ $<

%.$(STATIC_O): $(srcdir)/%.cpp
$(COMPILE.cc) $(STATICCPPFLAGS) $(STATICCXXFLAGS) -o $@ $<
%.o: $(srcdir)/%.cpp
$(COMPILE.cc) $(DYNAMICCPPFLAGS) $(DYNAMICCXXFLAGS) -o $@ $<

## Dependency rules
%.d: $(srcdir)/%.c
@echo "generating dependency information for $<"
@$(SHELL) -ec '$(GEN_DEPS.c) $< |
| sed '#s/\(.*\)/$@ : /g' | xargs rm -f $@'
%.d: $(srcdir)/%.cpp
@echo "generating dependency information for $<"
@$(SHELL) -ec '$(GEN_DEPS.cc) $< |
| sed '#s/\(.*\)/$@ : /g' | xargs rm -f $@'
## Versioned libraries rules

%.SO%.SO_TARGET_VERSION_MAJOR: %.SO%.SO_TARGET_VERSION
$(RM) $@ && ln -s ${<F} $@
%.SO: %.SO%.SO_TARGET_VERSION_MAJOR
$(RM) $@ && ln -s *F.SO%.SO_TARGET_VERSION $@

## Bind internal references

# LDflags that pkgdata will use
BIR_LDFLAGS= -Wl,-Bsymbolic

# Dependencies [i.e. map files] for the final library
BIR_DEPS=

## Remove shared library 's'
STATIC_PREFIX_WHEN_USED =
STATIC_PREFIX =

## End BSD-specific setup

1.1934 es-abstract 1.19.1

1.1934.1 Available under license:

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1937 jsreport-pdf-utils 1.10.1 1.10.1

1.1937.1 Available under license:
 MIT License

Copyright (c) 2017 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1938 set-value 2.0.1 2.0.1

1.1938.1 Available under license:
 The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1939 gson-fire 1.5.1

1.1939.1 Available under license:

Copyright 2015 Julio Viera

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1940 metrics 4.1.0

1.1940.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all...
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 MongoDB Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1941 asm-analysis 7.3.1

1.1941.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link SourceInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the { @link #SourceInterpreter(int)}
 * version.
 */

Found in path(s):
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/Interpreter.java
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/Analyzer.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-jar/org/objectweb/asm/tree/analysis/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link BasicInterpreter} for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the {@link #BasicInterpreter(int)}
 * version.
 */

Found in path(s):
* /opt/cola/permits/1135907546_1613628471.63/0/asm-analysis-7-3-1-sources-1-
  jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java

1.1942 duplexer 0.1.1 0.1.1
1.1942.1 Available under license :
Copyright (c) 2012 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1943 thrift 0.7.0
1.1943.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for the these subcomponents is subject to the terms and
conditions of the following licenses.

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

The following files contain some portions of code contributed under
the Thrift Software License (see doc/old-thrift-license.txt), and relicensed
under the Apache 2.0 License:

compiler/cpp/Makefile.am
compiler/cpp/src/generate/t_cocoa_generator.cc
compiler/cpp/src/generate/t_cpp_generator.cc
compiler/cpp/src/generate/t_csharp_generator.cc
compiler/cpp/src/generate/t_erl_generator.cc
compiler/cpp/src/generate/t_hs_generator.cc
compiler/cpp/src/generate/t_java_generator.cc
compiler/cpp/src/generate/t_ocaml_generator.cc
compiler/cpp/src/generate/t_perl_generator.cc
compiler/cpp/src/generate/t_php_generator.cc
compiler/cpp/src/generate/t_py_generator.cc
compiler/cpp/src/generate/t_rb_generator.cc
compiler/cpp/src/generate/t_st_generator.cc
compiler/cpp/src/generate/t_xsd_generator.cc
compiler/cpp/src/main.cc
compiler/cpp/src/parse/t_field.h
compiler/cpp/src/parse/t_program.h
compiler/cpp/src/platform.h
compiler/cpp/src/thriftl.ll
compiler/cpp/src/thrifti.yy
lib/csharp/src/Protocol/TBinaryProtocol.cs
lib/csharp/src/Protocol/TField.cs
lib/csharp/src/Protocol/TList.cs
lib/csharp/src/Protocol/TMap.cs
lib/csharp/src/Protocol/TMessage.cs
lib/csharp/src/Protocol/TMessageType.cs
lib/csharp/src/Protocol/TProtocol.cs
lib/csharp/src/Protocol/TProtocolException.cs
lib/csharp/src/Protocol/TProtocolFactory.cs
lib/csharp/src/Protocol/TProtocolUtil.cs
lib/csharp/src/Protocol/TSet.cs
lib/csharp/src/Protocol/TTStruct.cs
lib/csharp/src/Server/TServer.cs
lib/csharp/src/Server/TSimpleServer.cs
lib/csharp/src/Server/TTThreadPoolServer.cs
lib/csharp/src/TApplicationException.cs
lib/csharp/src/Thrift.csproj
lib/csharp/src/Thrift.sln
lib/csharp/src/TProcessor.cs
lib/csharp/src/Transport/TServerSocket.cs
lib/csharp/src/Transport/TServersTransport.cs
lib/csharp/src/Transport/TSocket.cs
lib/csharp/src/Transport/TStreamTransport.cs
lib/csharp/src/Transport/TTransport.cs
lib/csharp/src/Transport/TTransportException.cs
lib/csharp/src/Transport/TTransportFactory.cs
lib/csharp/ThriftMSBuildTask/Properties/AssemblyInfo.cs
lib/csharp/ThriftMSBuildTask/ThriftBuild.cs
lib/csharp/ThriftMSBuildTask/ThriftMSBuildTask.csproj
lib/rb/lib/thrift.rb
lib/st/README
lib/st/thrift.st
test/OptionalRequiredTest.cpp
test/OptionalRequiredTest.thrift
test/ThriftTest.thrift

--------------------------------------------------
For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

#  Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
#  Copying and distribution of this file, with or without
#  modification, are permitted in any medium without royalty provided
#  the copyright notice and this notice are preserved.

--------------------------------------------------
For the compiler/cpp/src/md5.[ch] components:

/*
Copyright (c) 1999, 2000, 2002 Aladdin Enterprises. All rights reserved.

This software is provided `as-is`, without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

L. Peter Deutsch
ghost@aladdin.com

*/

---------------------------------------------
For the lib/rb/setup.rb: Copyright (c) 2000-2005 Minero Aoki,
lib/ocaml/OCamlMakefile and lib/ocaml/README-OCamlMakefile components:
  Copyright (C) 1999 - 2007 Markus Mottl

Licensed under the terms of the GNU Lesser General Public License 2.1
(see doc/lgpl-2.1.txt for the full terms of this license)
This package was debianized by Thrift Developer's <dev@thrift.apache.org>.

This package and the Debian packaging is licensed under the Apache License,
see `/usr/share/common-licenses/Apache-2.0`.

The following informations was copied from Apache Thrift LICENSE file.

---------------------------------------------
SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for the these subcomponents is subject to the terms and
conditions of the following licenses.

---------------------------------------------
Portions of the following files are licensed under the MIT License:
lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

------------------------------------------------------------------------------

The following files contain some portions of code contributed under the Thrift Software License (see doc/old-thrift-license.txt), and relicensed under the Apache 2.0 License:

compiler/cpp/Makefile.am
compiler/cpp/src/generate/t_cocoa_generator.cc
compiler/cpp/src/generate/t_cpp_generator.cc
compiler/cpp/src/generate/t_csharp_generator.cc
compiler/cpp/src/generate/t_erl_generator.cc
compiler/cpp/src/generate/t hs_generator.cc
compiler/cpp/src/generate/t java_generator.cc
compiler/cpp/src/generate/t ocaml_generator.cc
compiler/cpp/src/generate/t_perl_generator.cc
compiler/cpp/src/generate/t php_generator.cc
compiler/cpp/src/generate/t_py_generator.cc
compiler/cpp/src/generate/t_rb_generator.cc
compiler/cpp/src/generate/t st_generator.cc
compiler/cpp/src/generate/t xsd_generator.cc
compiler/cpp/src/main.cc
compiler/cpp/src/parse/t_field.h
compiler/cpp/src/parse/t_program.h
compiler/cpp/src/platform.h
compiler/cpp/src/thriftl.ll
compiler/cpp/src/thrifti.yy
lib/csharp/src/Protocol/TBinaryProtocol.cs
lib/csharp/src/Protocol/TField.cs
lib/csharp/src/Protocol/TList.cs
lib/csharp/src/Protocol/TMap.cs
lib/csharp/src/Protocol/TMessage.cs
lib/csharp/src/Protocol/TMessageType.cs
lib/csharp/src/Protocol/TProtocol.cs
lib/csharp/src/Protocol/TProtocolException.cs
lib/csharp/src/Protocol/TProtocolFactory.cs
lib/csharp/src/Protocol/TProtocolUtil.cs
lib/csharp/src/Protocol/TSet.cs
lib/csharp/src/Protocol/TStruct.cs
lib/csharp/src/Protocol/TType.cs
lib/csharp/src/Server/TServer.cs
lib/csharp/src/Server/TSimpleServer.cs
lib/csharp/src/Server/ThreadPoolServer.cs
lib/csharp/src/TApplicationException.cs
lib/csharp/src/Thrift.csproj
lib/csharp/src/Thrift.sln
lib/csharp/src/TProcessor.cs
lib/csharp/src/Transport/TServerSocket.cs
lib/csharp/src/Transport/TServerTransport.cs
lib/csharp/src/Transport/TSocket.cs
lib/csharp/src/Transport/TSocketFactory.cs
lib/csharp/src/Transport/TransportException.cs
lib/csharp/src/Transport/TSocket.cs
lib/csharp/src/Transport/TSocketFactory.cs
lib/csharp/src/Transport/TransportFactory.cs
lib/csharp/ThriftMSBuildTask/Properties/AssemblyInfo.cs
lib/csharp/ThriftMSBuildTask/ThriftBuild.cs
lib/csharp/ThriftMSBuildTask/ThriftMSBuildTask.csproj
lib/rb/lib/thrift.rb
lib/st/README
lib/st/thrift.st
test/OptionalRequiredTest.cpp
test/OptionalRequiredTest.thrift
test/ThriftTest.thrift

For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

#   Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#   Copying and distribution of this file, with or without
#   modification, are permitted in any medium without royalty provided
#   the copyright notice and this notice are preserved.

For the compiler/cpp/src/md5.[ch] components:

/*
   Copyright (C) 1999, 2000, 2002 Aladdin Enterprises. All rights reserved.

   This software is provided 'as-is', without any express or implied
   warranty. In no event will the authors be held liable for any damages
   arising from the use of this software.

   Permission is granted to anyone to use this software for any purpose,
   including commercial applications, and to alter it and redistribute it
   freely, subject to the following restrictions:

   1. The origin of this software must not be misrepresented; you must not
      claim that you wrote the original software. If you use this software
      in a product, an acknowledgment in the product documentation would be
      appreciated but is not required.
   2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

L. Peter Deutsch
ghost@aladdin.com

*/

For the lib/rb/setup.rb: Copyright (c) 2000-2005 Minero Aoki,
lib/ocaml/OCamlMakefile and lib/ocaml/README-OCamlMakefile components:
Copyright (C) 1999 - 2007 Markus Mottl

Licensed under the terms of the GNU Lesser General Public License 2.1
(see doc/lgpl-2.1.txt for the full terms of this license)
Tue Oct 24 12:28:44 CDT 2006

Copyright (c) <2006> <Martin J. Logan, Erlware>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software (OTP Base, fslib, G.A.S) and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache Thrift
Copyright 2006-2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.1944 bcrypt-pbkdf 1.0.2 1.0.2
1.1944.1 Available under license:

The Blowfish portions are under the following license:

Blowfish block cipher for OpenBSD
Copyright 1997 Niels Provos <provos@physnet.uni-hamburg.de>
All rights reserved.

Implementation advice by David Mazieres <dm@lcs.mit.edu>.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The bcrypt_pbkdf portions are under the following license:

Copyright (c) 2013 Ted Unangst <tedu@openbsd.org>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Performance improvements (Javascript-specific):

Copyright 2016, Joyent Inc
Author: Alex Wilson <alex.wilson@joyent.com>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.1945 hawtjni-runtime 1.17
1.1945.1 Available under license:

Found license 'Eclipse Public License 1.0' in '* All rights reserved. This program and the accompanying materials *
are made available under the terms of the Eclipse Public License v1.0'

1.1946 libevent 1.4.13
1.1946.1 Available under license:

CMake - Cross Platform Makefile Generator
Copyright 2000-2013 Kitware, Inc.
Copyright 2000-2011 Insight Software Consortium
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of Kitware, Inc., the Insight Software Consortium,
nor the names of their contributors may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The above copyright and license notice applies to distributions of
CMake in source and binary form. Some source files contain additional
notices of original copyright by their contributors; see each source
for details. Third-party software packages supplied with CMake under
compatible licenses provide their own copyright notices documented in
corresponding subdirectories.

CMake was initially developed by Kitware with the following sponsorship:

* National Library of Medicine at the National Institutes of Health
  as part of the Insight Segmentation and Registration Toolkit (ITK).

* US National Labs (Los Alamos, Livermore, Sandia) ASC Parallel
  Visualization Initiative.

* National Alliance for Medical Image Computing (NAMIC) is funded by the
  National Institutes of Health through the NIH Roadmap for Medical Research,
  Grant U54 EB005149.

* Kitware, Inc.

Libevent is available for use under the following license, commonly known
as the 3-clause (or "modified") BSD license:

==============================================================================
Copyright (c) 2000-2007 Niels Provos <provos@citi.umich.edu>
Copyright (c) 2007-2012 Niels Provos and Nick Mathewson
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "$ AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=================================================================

Portions of Libevent are based on works by others, also made available by them under the three-clause BSD license above. The copyright notices are available in the corresponding source files; the license is as above. Here's a list:

log.c:
   Copyright (c) 2000 Dug Song <dugsong@monkey.org>
   Copyright (c) 1993 The Regents of the University of California.

strlcpy.c:
   Copyright (c) 1998 Todd C. Miller <Todd.Miller@courtesan.com>

win32select.c:
   Copyright (c) 2003 Michael A. Davis <mike@datanerds.net>

evport.c:
   Copyright (c) 2007 Sun Microsystems

ht-internal.h:
   Copyright (c) 2002 Christopher Clark

minheap-internal.h:
   Copyright (c) 2006 Maxim Yegorushkin <maxim.yegorushkin@gmail.com>

=================================================================
The arc4module is available under the following, sometimes called the "OpenBSD" license:

Copyright (c) 1996, David Mazieres <dm@uun.org>
Copyright (c) 2008, Damien Miller <djm@openbsd.org>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

===========================================

The Windows timer code is based on code from libutp, which is distributed under this license, sometimes called the "MIT" license.

Copyright (c) 2010 BitTorrent, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.1947 xmlbuilder 10.1.1 10.1.1
1.1947.1 Available under license :
The MIT License (MIT)

Copyright (c) 2013 Ozgur Ozcitak

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1948 levn 0.3.0 0.3.0
1.1948.1 Available under license:
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1949 kryo-shaded 3.0.3

1.1950 @babel/plugin-syntax-dynamic-import 7.8.3

1.1950.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1951 oauth-sign 0.9.0 0.9.0

1.1951.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole,
an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1953 batik-xml 1.13

1.1953.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878576_1616835470.15/0/batik-xml-1-13-sources-1-jar/org/apache/batik/xml/XMLStreamNormalizingReader.java
* /opt/cola/permits/1146878576_1616835470.15/0/batik-xml-1-13-sources-1-jar/org/apache/batik/xml/LexicalUnits.java
* /opt/cola/permits/1146878576_1616835470.15/0/batik-xml-1-13-sources-1-jar/org/apache/batik/xml/XMLUtilities.java
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1954 mockito-core 2.23.4

1.1954.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1955 babel-plugin-transform-es2015-for-of
6.23.0 6.23.0
1.1955.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1956 jackson 2.12.1

1.1956.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.1957 mongodb-driver-core 4.2.3

1.1957.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2017 Tom Bentley
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
  jar/com/mongodb/internal/authentication/SaslPrep.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2005 Brian Goetz and Tim Peierls
 * Released under the Creative Commons Attribution License
 * (http://creativecommons.org/licenses/by/2.5)
 * Official home: http://www.jcip.net
 *
 * Any republication or derived work distributed in source code form
 * must include this copyright and license notice.
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
  jar/com/mongodb/annotations/NotThreadSafe.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
  jar/com/mongodb/annotations/Immutable.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/annotations/ThreadSafe.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original Work: MIT License, Copyright (c) [2015-2020] all contributors
 * https://github.com/marianobarrios/tls-channel
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/BufferHolder.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/BufferAllocator.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/TlsChannelFlowControlException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/SniSslContextFactory.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/HeapBufferAllocator.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/WouldBlockException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/ClientTlsChannel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/NeedsReadException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/TlsChannelImpl.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/ServerTlsChannel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/async/package-info.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/NeedsTaskException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/package-info.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/TlsChannelBuilder.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferSet.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/util/DirectBufferDeallocator.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/TlsChannelCallbackException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/async/ExtendedAsynchronousByteChannel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/ByteBufferUtil.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/DirectBufferAllocator.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/TrackingAllocator.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/async/AsynchronousTlsChannelGroup.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/Util.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/TlsChannel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/impl/TlsExplorer.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/tlschannel/NeedsWriteException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008-present MongoDB, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/

/**
 * Returns an array containing all of the elements in this deque, in
 * proper sequence (from first to last element).
 *
 * <p>The returned array will be "safe" in that no references to it are
 * maintained by this deque. (In other words, this method must allocate
 * a new array). The caller is thus free to modify the returned array.
 *
 * This method acts as bridge between array-based and collection-based
 * APIs.
 *
 * @return an array containing all of the elements in this deque
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
  jar/com/mongodb/internal/connection/ConcurrentLinkedDeque.java
No license file was found, but licenses were detected in source scan.

/ *
 * Copyright 2008-present MongoDB, Inc.
 * Copyright 2010 The Guava Authors
 * Copyright 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
  jar/com/mongodb/annotations/Beta.java
No license file was found, but licenses were detected in source scan.

/ *
 * Copyright 2008-present MongoDB, Inc.
 *
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Open Source Used In DNAC Platform Artic
Open Source Used In DNAC Platform Arica 9093
jar/com/mongodb/client/model/Sorts.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/operation/TransactionOperation.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/bulk/package-info.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/client/model/ReturnDocument.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/client/model/UpdateOptions.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/client/model/geojson/codecs/MultiPointCodec.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/connection/AsynchronousSocketChannelStreamFactory.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/ServerMonitor.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/client/model/UpdateManyModel.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/BSONTimestampCodec.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/authentication/NativeAuthenticationHelper.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/BulkWriteBatchCombiner.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/MongoIncompatibleDriverException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/Server.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/MongoCursorNotFoundException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/BasicDBObjectFactory.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/binding/AsyncReadBinding.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/ErrorCategory.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/UpdateProtocol.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/operation/ChangeStreamBatchCursor.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/MongoSecurityException.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-
jar/com/mongodb/internal/connection/AuthorizationHeader.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Copyright (c) 2008-2014 Atlassian Pty Ltd
*/
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/assertions/Assertions.java
No license file was found, but licenses were detected in source scan.

 /*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/ReadPreferenceHedgeOptions.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/lang/NonNullApi.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/internal/operation/WriteConcernHelper.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/lang/Nullable.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/internal/async/client/AsyncClientSession.java
* /opt/cola/permits/1202446273_1631568031.75/0/mongodb-driver-core-4-2-3-sources-jar/com/mongodb/internal/connection/ReadConcernHelper.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 * Original Work: MIT License, Copyright (c) [2015-2018] all contributors
 * https://github.com/marianobarrios/tls-channel
 */

1.1958 babel-plugin-syntax-exponentiation-operator 6.13.0 6.13.0

1.1958.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1959 filename-regex 2.0.1

1.1959.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1960 react-intl 2.4.0 2.4.0

1.1960.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1961 is-extglob 2.1.1

#### 1.1961.1 Available under license :  
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.1962 ms 2.1.1

#### 1.1962.1 Available under license :  
The MIT License (MIT)

Copyright (c) 2016 Zeit, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1963 mongodb-core 2.1.20 2.1.20

1.1963.1 Available under license:

Mongodb-core uses third-party libraries or other resources that may be distributed under licenses different than the Mongodb-core software.

In the event that we accidentally failed to list a required notice, please bring it to our attention through any of the ways detailed here:

    mongodb-dev@googlegroups.com

The attached notices are provided for information only.

For any licenses that require disclosure of source, sources are available at https://github.com/mongodb/node-mongodb-native.

1) License Notice for nan.h
-------------------------------

The MIT License (MIT)
=====================

Copyright (c) 2016 NAN contributors
-----------------------------------

NAN contributors listed at <https://github.com/nodejs/nan#contributors>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1964 extsprintf 1.3.0 1.3.0

1.1964.1 Available under license:

Copyright (c) 2012, Joyent, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1965 @babel/helper-module-imports 7.16.0

1.1965.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1966 jsreport-jsrender 2.1.1 2.1.1

1.1966.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object

code under terms of your choice, provided that, if the incorporated

material is not limited to numerical parameters, data structure

layouts and accessors, or small macros, inline functions and templates

(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the

Library is used in it and that the Library and its use are

covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the

portions of the Library contained in the Combined Work and reverse

engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that

the Library is used in it and that the Library and its use are

covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during

execution, include the copyright notice for the Library among

these notices, as well as a reference directing the user to the

copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this

License, and the Corresponding Application Code in a form

suitable for, and under terms that permit, the user to

recombine or relink the Application with a modified version of

the Linked Version to produce a modified Combined Work, in the

manner specified by section 6 of the GNU GPL for conveying

Corresponding Source.

1) Use a suitable shared library mechanism for linking with the

Library. A suitable mechanism is one that (a) uses at run time

a copy of the Library already present on the user's computer

system, and (b) will operate properly with a modified version

of the Library that is interface-compatible with the Linked
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.1967 swagger 1.5.0

1.1967.1 Available under license:
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1433634398680
Build-Jdk: 1.7.0_72
Built-By: tony
Bundle-Description: Sonatype helps open source projects to set up Maven repositories on https://oss.sonatype.org/
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.html
Bundle-ManifestVersion: 2
Bundle-Name: swagger-models
Bundle-SymbolicName: io.swagger.models
Bundle-Version: 1.5.0
Created-By: Apache Maven Bundle Plugin
implementation-version: 1.5.0
Import-Package: com.fasterxml.jackson.annotation;version="[2.4,3)";javax.xml.bind.annotation,org.slf4j;version="[1.6,2)"
mode: development
package: io.swagger
Require-Capability: osgi.ee;filter:="(&osgi.ee=JavaSE)(version=1.6)"
Tool: Bnd-2.3.0.201405100607
url: https://github.com/swagger-api/swagger-core/modules/swagger-models

Found in path(s):
* /opt/cola/permits/1001159702_1607016290.96/0/swagger-models-1-5-0.jar/META-INF/MANIFEST.MF

1.1968 d-bus 1.10.6-1ubuntu3.5

1.1968.1 Available under license:
D-Bus is licensed to you under your choice of the Academic Free License version 2.1, or the GNU General Public License version 2 (or, at your option any later version).
Both licenses are included here. Some of the standalone binaries are under the GPL only; in particular, but not limited to, tools/dbus-cleanup-sockets.c and test/decode-gcov.c. Each source code file is marked with the proper copyright information - if you find a file that isn't marked please bring it to our attention.

The Academic Free License
v. 2.1

This Academic Free License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following notice immediately following the copyright notice for the Original Work:

Licensed under the Academic Free License version 2.1

1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:

a) to reproduce the Original Work in copies;

b) to prepare derivative works ("Derivative Works") based upon the Original Work;

c) to distribute copies of the Original Work and Derivative Works to the public;

d) to perform the Original Work publicly; and

e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license, under patent claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.
4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to license.

5) This section intentionally omitted.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and
all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries, and by international treaty. Therefore, by exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, against Licensor or any licensee alleging that the Original Work infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware.

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. 101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys' fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.
13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.

This license is Copyright (C) 2003-2004 Lawrence E. Rosen. All rights reserved. Permission is hereby granted to copy and distribute this license without modification. This license may not be modified without the express written permission of its copyright owner.

--

END OF ACADEMIC FREE LICENSE. The following is intended to describe the essential differences between the Academic Free License (AFL) version 1.0 and other open source licenses:

The Academic Free License is similar to the BSD, MIT, UoI/NCSA and Apache licenses in many respects but it is intended to solve a few problems with those licenses.

* The AFL is written so as to make it clear what software is being licensed (by the inclusion of a statement following the copyright notice in the software). This way, the license functions better than a template license. The BSD, MIT and UoI/NCSA licenses apply to unidentified software.

* The AFL contains a complete copyright grant to the software. The BSD and Apache licenses are vague and incomplete in that respect.

* The AFL contains a complete patent grant to the software. The BSD, MIT, UoI/NCSA and Apache licenses rely on an implied patent license and contain no explicit patent grant.
* The AFL makes it clear that no trademark rights are granted to the licensor's trademarks. The Apache license contains such a provision, but the BSD, MIT and UoI/NCSA licenses do not.

* The AFL includes the warranty by the licensor that it either owns the copyright or that it is distributing the software under a license. None of the other licenses contain that warranty. All other warranties are disclaimed, as is the case for the other licenses.

* The AFL is itself copyrighted (with the right granted to copy and distribute without modification). This ensures that the owner of the copyright to the license will control changes. The Apache license contains a copyright notice, but the BSD, MIT and UoI/NCSA licenses do not.

--

START OF GNU GENERAL PUBLIC LICENSE

--

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.1969 jetty-servlets 9.3.27.v20190418

1.1970 netty-transport 4.0.48.Final

1.1970.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Handles or intercepts a {@link ChannelInboundInvoker} or {@link ChannelOutboundInvoker} operation, and
 * forwards it
 * to the next handler in a {@link ChannelPipeline}.
 *
 * <h3>Sub-types</h3>
 * <p>
 * [@link ChannelHandler] itself does not provide many methods. To handle a
 * a {@link ChannelInboundInvoker} or {@link ChannelOutboundInvoker} operation
 * you need to implement its sub-interfaces. There are many different sub-interfaces
 * which handles inbound and outbound operations.
 *
 * But the most useful for developers may be:
 * <ul>
 * <li>[@link ChannelInboundHandlerAdapter] handles and intercepts inbound operations</li>
 * <li>[@link ChannelOutboundHandlerAdapter] handles and intercepts outbound operations</li>
 * </ul>
 *
 * You will also find more detailed explanation from the documentation of
 * each sub-interface on how an event is interpreted when it goes upstream and
 * downstream respectively.
 *
 * <h3>The context object</h3>
A `ChannelHandler` is provided with a `ChannelHandlerContext` object. A `ChannelHandler` is supposed to interact with the `ChannelPipeline` it belongs to via a context object. Using the context object, the `ChannelHandler` can pass events upstream or downstream, modify the pipeline dynamically, or store the information (using `AttributeKey`) which is specific to the handler.

<h3>State management</h3>

A `ChannelHandler` often needs to store some stateful information. The simplest and recommended approach is to use member variables:

```java
public interface Message {
    // your methods here
}

public class DataServerHandler extends SimpleChannelInboundHandler<Message> {

    private boolean loggedIn;

    @Override
    public void channelRead0(ChannelHandlerContext ctx, Message message) {
        Channel ch = e.getChannel();
        if (message instanceof LoginMessage) {
            authenticate((LoginMessage) message);
            loggedIn = true;
        } else if (message instanceof GetDataMessage) {
            if (loggedIn) {
                ch.write(fetchSecret((GetDataMessage) message));
            } else {
                fail();
            }
        }
    }
    ...
}
```

Because the handler instance has a state variable which is dedicated to one connection, you have to create a new handler instance for each new channel to avoid a race condition where a unauthenticated client can get the confidential information:

```java
// Create a new handler instance per channel.
// See {ChannelInitializer#initChannel(Channel)}.
public class DataServerInitializer extends ChannelInitializer<Channel> {

    @Override
    public void initChannel(Channel channel) {

    }
}
```
* channel.pipeline().addLast("handler", new DataServerHandler());
* }
* }
* }
* </pre>
* 
* <h4>Using {@link AttributeKey}</h4>
* 
* Although it's recommended to use member variables to store the state of a
* handler, for some reason you might not want to create many handler instances.
* In such a case, you can use [@link AttributeKey]s which is provided by
* [@link ChannelHandlerContext]:
* <pre>
* public interface Message {
*     // your methods here
* }
* *
* [@code @Sharable]
* public class DataServerHandler extends [@link SimpleChannelInboundHandler]&lt;Message&gt; {
*     private final [@link AttributeKey]&lt;@link Boolean&gt; auth =
*         @link AttributeKey#valueOf(String) AttributeKey.valueOf("auth");
* *
*     [@code @Override]
*     public void channelRead([@link ChannelHandlerContext] ctx, Message message) {
*         @link Attribute&lt;@link Boolean&gt; attr = ctx.attr(auth);
*         @link Channel ch = ctx.channel();
*         if (message instanceof LoginMessage) {
*             authenticate((LoginMessage) o);
*             attr.set(true);
*         } else (message instanceof GetDataMessage) {
*             if (Boolean.TRUE.equals(attr.get())) {
*                 ch.write(fetchSecret((GetDataMessage) o));
*             } else {
*                 fail();
*             }
*         }
*         }
*         ...
*     }
* </pre>
* Now that the state of the handler is attached to the [@link ChannelHandlerContext], you can add the
* same handler instance to different pipelines:
* <pre>
* public class DataServerInitializer extends [@link ChannelInitializer]&lt;@link Channel&gt; {
*     private static final DataServerHandler <b>SHARED</b> = new DataServerHandler();
*     
*     [@code @Override]
*     public void init([@link Channel channel]) {
*         channel.pipeline().addLast("handler", <b>SHARED</b>);
*     }
* </pre>
public void initChannel(@link Channel} channel) {
    channel.pipeline().addLast("handler", \<b\>SHARED</b>);
}
</pre>

<h4>The \{code @Sharable\} annotation</h4>

In the example above which used an \{link AttributeKey\},
you might have noticed the \{code @Sharable\} annotation.

If a \{link ChannelHandler\} is annotated with the \{code @Sharable\}
anotation, it means you can create an instance of the handler just once and
add it to one or more \{link ChannelPipeline\}s multiple times without
a race condition.

If this annotation is not specified, you have to create a new handler
every time you add it to a pipeline because it has unshared state
such as member variables.

This annotation is provided for documentation purpose, just like

Additional resources worth reading</h3>

Please refer to the \{link ChannelHandler\}, and
\{link ChannelPipeline\} to find out more about inbound and outbound operations,
what fundamental differences they have, how they flow in a pipeline, and how to handle
the operation in your application.

/\n
Found in path(s):
LOPT/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/ChannelHandler.java

No license file was found, but licenses were detected in source scan.

/\n
Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/PendingWriteQueue.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/DefaultAddressedEnvelope.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/ChannelProgressiveFutureListener.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/bootstrap/ChannelFactory.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/ChannelHandlerAdapter.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/ChannelOutboundBuffer.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/ChannelProgressivePromise.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/DefaultMessageSizeEstimator.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/ConnectTimeoutException.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/socket/oio/OioSocketChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
  jar/io/netty/channel/DefaultChannelProgressivePromise.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/oio/OioByteStreamChannel.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/CombinedChannelDuplexHandler.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/ChannelFutureListener.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/socket/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/socket/nio/NioSocketChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/socket/nio/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/ChannelPipelineException.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/local/LocalChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/group/ChannelGroupFuture.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/socket/DefaultDatagramChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/bootstrap/AbstractBootstrap.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/socket/nio/NioDatagramChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/ChannelException.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/CompleteChannelFuture.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/ChannelDuplexHandler.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/VoidChannelPromise.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/ChannelOutboundHandler.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/ChannelInboundHandlerAdapter.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/oio/OioEventLoopGroup.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/ChannelOption.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/oio/OioEventLoopGroup.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/oio/AbstractOioByteChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/local/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/oio/OioSocketChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/nio/NioServerSocketChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/DefaultFileRegion.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/SocketChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/oio/OioServerSocketChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/oio/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/local/LocalEventLoop.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/FixedRecvByteBufAllocator.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/AdaptiveRecvByteBufAllocator.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/AbstractChannelHandlerContext.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/group/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/nio/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/local/LocalEventLoop.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/nio/NettyChannelPipeline.java
jar/io/netty/channel/DefaultChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/nio/AbstractNioChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/local/LocalChannelRegistry.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/ChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/EventLoopGroup.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/embedded/EmbeddedChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/DefaultChannelPromise.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/local/LocalServerChannel.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/EventLoopException.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/ChannelFuture.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/embedded/package-info.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/DatagramChannelConfig.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/socket/DatagramChannel.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-
jar/io/netty/channel/DefaultChannelHandlerContext.java

No license file was found, but licenses were detected in source scan.
~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/META-INF/maven/io.netty/netty-transport/pom.xml
No license file was found, but licenses were detected in source scan.

/
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4-0-48-final-sources-jar/io/netty/channel/nio/SelectedSelectionKeySetSelector.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/ChannelPoolHandler.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/FixedChannelPool.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/ChannelPool.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/AbstractChannelPoolMap.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/AbstractChannelPoolHandler.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/ChannelPoolMap.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/ChannelHealthChecker.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/SimpleChannelPool.java
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
  jar/io/netty/channel/pool/package-info.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446789_1631568416.78/0/netty-transport-4.0.48-final-sources-
1.1971 kerb-identity 1.0.0

1.1971.1 Available under license:

Kerby-kerb Identity
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation.
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.1972 parquet-common 1.12.0

1.1972.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/LittleEndianDataOutputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/CapacityByteArrayOutputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/hadoop/metadata/Canonicalizer.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/HeapByteBufferAllocator.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/Log.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/ParquetRuntimeException.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/hadoop/metadata/ColumnPath.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/LittleEndianDataInputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/BytesUtils.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/Preconditions.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/bytes/DirectByteBufferAllocator.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/Ints.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/IOExceptionUtils.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/SemanticVersion.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/BytesInput.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/ShouldNeverHappenException.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/SingleBufferInputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/glob/GlobNode.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/glob/GlobParser.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/Files.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/glob/GlobExpander.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/util/DynConstructors.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/util/DynMethods.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/Strings.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/ByteBufferInputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/glob/WildcardPath.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/ByteBufferAllocator.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/ConcatenatingByteArrayCollector.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/VersionParser.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/Closeables.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
*/
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/PositionOutputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/InputFile.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/OutputFile.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/hadoop/codec/CompressionCodecNotSupportedException.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/compression/CompressionCodecFactory.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/SeekableInputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/hadoop/metadata/CompressionCodecName.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/Exceptions.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/DelegatingPositionOutputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/io/DelegatingSeekableInputStream.java
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-
jar/org/apache/parquet/bytes/MultiBufferInputStream.java

No license file was found, but licenses were detected in source scan.

/**
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* <p/>
* http://www.apache.org/licenses/LICENSE-2.0
* <p/>
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446201_1631568151.58/0/parquet-common-1-12-0-sources-jar/org/apache/parquet/OutputStreamCloseException.java

1.1973 ncp 2.0.0 2.0.0

1.1973.1 Available under license:

# MIT License

### Copyright (C) 2011 by Charlie McConnell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1974 asm-all-repackaged 2.2.0-b21

1.1974.1 Available under license:

No license file was found, but licenses were detected in source scan.

ASM XML Adapter
Copyright (c) 2004, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/asm-xml.dtd

No license file was found, but licenses were detected in source scan.

/***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2005 INRIA, France Telecom
* All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* ***/
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/commons/InstructionAdapter.java
No license file was found, but licenses were detected in source scan.

2011 Oracle and/or its affiliates. All rights reserved.

The contents of this file are subject to the terms of either the GNU
General Public License Version 2 only ("GPL") or the Common Development
and Distribution License("CDDL") (collectively, the "License"). You
may not use this file except in compliance with the License. You can
obtain a copy of the License at
https://glassfish.dev.java.net/public/CDDL+GPL_1_1.html
or packager/legal/LICENSE.txt. See the License for the specific
language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each
file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath"
exception as provided by Oracle in the GPL Version 2 section of the License
file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields
enclosed by brackets [] replaced by your own identifying information:
"Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or
only the GPL Version 2, indicate your decision by adding "[Contributor]
elects to include this software in this distribution under the [CDDL or GPL
Version 2] license." If you don't indicate a single choice of license, a
recipient has the option to distribute your version of this file under
either the CDDL, the GPL Version 2 or to extend the choice of license to
its licensees as provided above. However, if you add GPL Version 2 code
and therefore, elected the GPL Version 2 license, then the option applies
only if the new code is made subject to such option by the copyright
holder.

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/pom.xml
No license file was found, but licenses were detected in source scan.

/****
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/Type.java
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/commons/GeneratorAdapter.java
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
* Redistribution and use in source and binary forms, with or without
  * modification, are permitted provided that the following conditions
  * are met:
  * 1. Redistributions of source code must retain the above copyright
     * notice, this list of conditions and the following disclaimer.
  * 2. Redistributions in binary form must reproduce the above copyright
     * notice, this list of conditions and the following disclaimer in the
     * documentation and/or other materials provided with the distribution.
  * 3. Neither the name of the copyright holders nor the names of its
     * contributors may be used to endorse or promote products derived from
     * this software without specific prior written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
  * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
  * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
  * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
  * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
  * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
  * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
  * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
  * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
  * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
  * THE POSSIBILITY OF SUCH DAMAGE.
  */

Found in path(s):
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/SAXFieldAdapter.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/SAXAdapter.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/SAXClassAdapter.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/SAXCodeAdapter.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/ASMContentHandler.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/xml/SAXAnnotationAdapter.java
  * /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-jar/org/objectweb/asm/XML/Processor.java

No license file was found, but licenses were detected in source scan.

2005 INRIA, France Telecom
* All rights reserved.
 *
  * Redistribution and use in source and binary forms, with or without
  * modification, are permitted provided that the following conditions
  * are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/tree/analysis/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/commons/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/tree/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/signature/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/util/package.html
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
  jar/org/objectweb/asm/attrs/package.html
No license file was found, but licenses were detected in source scan.

* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1136447751_1613767073.0/0/asm-all-repackaged-2-2-0-b21-sources-3-
jar/org/objectweb/asm/xml/package.html
No license file was found, but licenses were detected in source scan.

/**
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such
combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.
6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly
stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work...
or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core
The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.
org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.
1.1977 is-callable 1.2.3
1.1977.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1978 har-validator 5.1.5 5.1.5
1.1978.1 Available under license:
MIT License

Copyright (c) 2018 Ahmad Nassri <ahmad@ahmadnassri.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1979 aggdesigner-algorithm 6.0

1.1980 topo 1.1.0 1.1.0

1.1980.1 Available under license:

Copyright (c) 2012-2014, Walmart and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE
FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/topo/graphs/contributors

1.1981 path-is-absolute 1.0.1 1.0.1

1.1981.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1982 jersey-mvc 2.25.1

1.1982.1 Available under license:

Found license 'General Public License 2.0' in '* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License' 

Found license 'General Public License 2.0' in '# Copyright (c) 2013-2014 Oracle and/or its affiliates. All rights reserved. # The contents of this file are subject to the terms of either the GNU # General Public License Version 2 only ("GPL") or the Common Development # and Distribution License("CDDL") (collectively, the "License"). You # may not use this file except in compliance with the License. You can # Oracle designates this particular file as subject to the "Classpath" # exception as provided by Oracle in the GPL Version 2 section of the License' 

Found license 'General Public License 2.0' in '* Copyright (c) 2010-2014 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License' 

Found license 'General Public License 2.0' in '* Copyright (c) 2012-2015 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'
Found license 'General Public License 2.0' in "Copyright (c) 2010-2015 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License’

1.1983 babel-plugin-polyfill-regenerator 0.3.0
1.1983.1 Available under license:
MIT License

Copyright (c) 2014-present Nicol Ribaudo and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1984 socket.io-adapter 0.5.0 0.5.0
1.1984.1 Available under license:
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1985 jetty-util 9.4.41.v20210516

1.1985.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

   a) it complies with the terms and conditions of this Agreement; and
   b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and

b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.
For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

==============================================================

Jetty Web Container

Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

==============================================================
The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.
* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.1986 merge-class-names 1.3.0 1.3.0

1.1986.1 Available under license:

MIT License

Copyright (c) 2017-2019 Wojciech Maj

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2016 Alex Indigo

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

The FreeType Project LICENSE

----------------------------

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction
============

The FreeType Project is distributed in several archive packages;
some of them may contain, in addition to the FreeType font engine,
various tools and contributions which rely on, or relate to, the
FreeType Project.
This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be interested in any kind of bug reports. (`as is' distribution)

- You can use this software for whatever you want, in parts or full form, without having to pay us. (`royalty-free' usage)

- You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. (`credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

````
... Portions of this software are copyright <year> The FreeType Project (www.freetype.org). All rights reserved.
...```

Please replace `<year>` with the value from the FreeType version you actually use.

Legal Terms
==========

0. Definitions
--------------

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally
distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the 'FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty
---------------

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution
-----------------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.
Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising
---------------

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts
-----------

There are two mailing lists related to FreeType:

- freetype@nongnu.org

  Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

- freetype-devel@nongnu.org

  Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.
Our home page can be found at

https://www.freetype.org

--- end of FTL.TXT ---
# Files that don't get a copyright, or which are taken from elsewhere.
#
# All lines in this file are patterns, including the comment lines; this
# means that e.g. `FTL.TXT' matches all files that have this string in
# the file name (including the path relative to the current directory,
# always starting with `./').
#
# Don't put empty lines into this file!
#
.gitignore
#
bUILDS/unIX/pKG.m4
#
docs/FTL.TXT
docs/GPLv2.TXT
#
include/freetype/internal/fthash.h
#
sRC/base/fthash.c
src/base/md5.c
src/base/md5.h
#
src/bdf/bdf.c
src/bdf/bdf.h
src/bdf/bdfdrivr.c
src/bdf/bdfdrivr.h
src/bdf/bdferror.h
src/bdf/bdflib.c
src/bdf/module.mk
src/bdf/README
src/bdf/rules.mk
#
src/pcf/module.mk
src/pcf/pcf.c
src/pcf/pcf.h
src/pcf/pcfdrivr.c
src/pcf/pcfdrivr.h
src/pcf/pcferror.h
src/pcf/pcfread.c
src/pcf/pcfutil.c
src/pcf/pcfutil.h
src/pcf/README
src/pcf/rules.mk
#
src/gzip/adler32.c
src/gzip/infblock.c
src/gzip/infblock.h
src/gzip/infcodes.c
src/gzip/infcodes.h
src/gzip/inffixed.h
src/gzip/inflate.c
src/gzip/inftrees.c
src/gzip/inftrees.h
src/gzip/infutil.c
src/gzip/infutil.h
src/gzip/zconf.h
src/gzip/zlib.h
src/gzip/zutil.c
src/gzip/zutil.h
#
src/tools/apinames.c
src/tools/ftrandom/ftrandom.c
#
# EOF

The FreeType 2 font engine is copyrighted work and cannot be used legally without a software license. In order to make this project usable to a vast majority of developers, we distribute it under two mutually exclusive open-source licenses.

This means that you must choose one of the two licenses described below, then obey all its terms and conditions when using FreeType 2 in any of your projects or products.

- The FreeType License, found in the file `FTL.TXT', which is similar to the original BSD license *with* an advertising clause that forces you to explicitly cite the FreeType project in your product's documentation. All details are in the license file. This license is suited to products which don't use the GNU General Public License.

  Note that this license is compatible to the GNU General Public License version 3, but not version 2.

- The GNU General Public License version 2, found in `GPLv2.TXT' (any later version can be used also), for programs which already use the GPL. Note that the FTL is incompatible with GPLv2 due to its advertisement clause.
The contributed BDF and PCF drivers come with a license similar to that of the X Window System. It is compatible to the above two licenses (see file src/bdf/README and src/pcf/README). The same holds for the files `fthash.c' and `fthash.h'; their code was part of the BDF driver in earlier FreeType versions.

The gzip module uses the zlib license (see src/gzip/zlib.h) which too is compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds) is in the public domain.

--- end of LICENSE.TXT ---

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable
      source code, which must be distributed under the terms of Sections
      1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three
      years, to give any third party, for a charge no more than your
      cost of physically performing source distribution, a complete
      machine-readable copy of the corresponding source code, to be
      distributed under the terms of Sections 1 and 2 above on a medium
      customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer
      to distribute corresponding source code. (This alternative is
      allowed only for noncommercial distribution and only if you
      received the program in object code or executable form with such
      an offer, in accord with Subsection b above.)

   The source code for a work means the preferred form of the work for
   making modifications to it.  For an executable work, complete source
   code means all the source code for all modules it contains, plus any
   associated interface definition files, plus the scripts used to
   control compilation and installation of the executable.  However, as a
   special exception, the source code distributed need not include
   anything that is normally distributed (in either source or binary
   form) with the major components (compiler, kernel, and so on) of the
   operating system on which the executable runs, unless that component
   itself accompanies the executable.

   If distribution of executable or object code is made by offering
   access to copy from a designated place, then offering equivalent
   access to copy the source code from the same place counts as
   distribution of the source code, even though third parties are not
   compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
   except as expressly provided under this License.  Any attempt
   otherwise to copy, modify, sublicense or distribute the Program is
   void, and will automatically terminate your rights under this License.
   However, parties who have received copies, or rights, from you under
   this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING FROM THE USE OR INABILITY TO USE THE PROGRAM OR FROM OTHER DEALINGS IN THE PROGRAM.
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year  name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.1989 syslog-java-client 1.1.7

1.1990 async 2.6.3 2.6.3
1.1990.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.1991 mkdirp 0.3.0 0.3.0

1.1991.1 Available under license:

Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1992 http-errors 1.6.2 1.6.2

1.1992.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com
Copyright (c) 2016 Douglas Christopher Wilson doug@somethingdoug.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE POI SUBCOMPONENTS:

Apache POI includes subcomponents with separate copyright notices and license terms. Your use of these subcomponents is subject to the terms and conditions of the following licenses:

Office Open XML schemas (poi-ooxml-full-*.jar)

The Office Open XML schema definitions used by Apache POI are a part of the Office Open XML ECMA Specification (ECMA-376, [1]). As defined in section 9.4 of the ECMA bylaws [2], this specification is available to all interested parties without restriction:

9.4 All documents when approved shall be made available to all interested parties without restriction.

Furthermore, both Microsoft and Adobe have granted patent licenses to this work [3,4,5].

Bouncy Castle library (bcprov-*.jar, bcpg-*.jar, bcpkix-*.jar)

Copyright (c) 2000 - 2021 The Legion of the Bouncy Castle Inc.
(https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CurvesAPI / Curve API

BSD License

Copyright (c) 2000-2015 www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

SLF4J library (slf4j-api-*.jar)

Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

inbot-utils (https://github.com/Inbot/inbot-utils)

The MIT License (MIT)

Copyright (c) 2015 Inbot

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache POI
Copyright 2003-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This product contains parts that were originally based on software from BEA.
Copyright (c) 2000-2003, BEA Systems, <http://www.bea.com/> (dead link),
which was acquired by Oracle Corporation in 2008.
<https://en.wikipedia.org/wiki/BEA_Systems>

This product contains W3C XML Schema documents. Copyright 2001-2003 (c)
World Wide Web Consortium (Massachusetts Institute of Technology, European
Research Consortium for Informatics and Mathematics, Keio University)

This product contains the chunks_parse_cmds.tbl file from the vsdump program.
Copyright (C) 2006-2007 Valek Filippov (frob@df.ru)

This product contains parts of the eID Applet project
Copyright (c) 2009-2018
FedICT (federal ICT department of Belgium), e-Contract.be BVBA (https://www.e-contract.be),
Bart Hanssens from FedICT

1.1994 path-to-regexp 0.1.7 0.1.7
1.1994.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is

Open Source Used In DNAC Platform Arica 9205
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1995 ajv 4.11.8

1.1995.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.1996 amqp-client 4.8.0

1.1996.1 Available under license :
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see Copyright (c) 2007-2016 Pivotal Software, Inc. All Rights Reserved Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may
obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS, * Based on org.stringtree.json.JSONWriter, licensed under APL and'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see Copyright (c) 2007-2016 Pivotal Software, Inc. All Rights Reserved Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS, * Based on org.stringtree.json.JSONReader, licensed under APL and'
Found license 'General Public License 2.0' in '// Copyright (c) 2018 Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see *
Expert Group and released to the public domain, as explained at'
Found license 'General Public License 2.0' in '// Copyright (c) 2018-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see * more details.'
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see final static String LICENSE="Licensed under the MPL. See http://www.rabbitmq.com/";
Found license 'General Public License 2.0' in '// Copyright (c) 2007-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'
Found license 'General Public License 2.0' in '// Copyright (c) 2017-Present Pivotal Software, Inc. All rights reserved. // This software, the RabbitMQ Java client library, is triple-licensed under the // Mozilla Public License 1.1 ("MPL"), the GNU General Public License version 2 // ("GPL") and the Apache License version 2 ("ASL"). For the MPL, please see'

1.1997 arr-union 3.1.0 3.1.0
1.1997.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.1998 highlight-words-core 1.2.2 1.2.2

1.1998.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Treasure Data

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.1999 yallist 2.1.2 2.1.2
1.1999.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2000 java-sizeof 0.0.5

1.2000.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * This class contains black magic stuff based mostly on proprietary
 * APIs and hacks. Use at your own risk.
 */

Found in path(s):
* /opt/cola/permits/1258272178_1643050051.82/0/java-sizeof-0-0-5-sources-jar/com/carrotsearch/sizeof/BlackMagic.java

No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.

Open Source Used In DNAC Platform Arica 9209
Found in path(s):
* /opt/cola/permits/1258272178_1643050051.82/0/java-sizeof-0-0-5-sources-jar/com/carrotsearch/sizeof/Constants.java
* /opt/cola/permits/1258272178_1643050051.82/0/java-sizeof-0-0-5-sources-jar/com/carrotsearch/sizeof/RamUsageEstimator.java

1.2001 babel-register 6.26.0

1.2001.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2002 fsevents 2.1.3 2.1.3
1.2002.1 Available under license :
MIT License
---------

Copyright (C) 2010-2020 by Philipp Dunkel, Ben Noordhuis, Elan Shankar, Paul Miller

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2003 json-refs 3.0.15 3.0.15
1.2003.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Jeremy Whitlock
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2004 strip-ansi 6.0.1

1.2004.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2005 deep-extend 0.6.0
1.2005.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013-2018, Viacheslav Lotsmanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2006 node-releases 2.0.1
1.2006.1 Available under license:
The MIT License

Copyright (c) 2017 Sergey Rubanov (https://github.com/chicoxyzzy)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2007 @babel/core 7.16.0
1.2007.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2008 traverse 0.3.9 0.3.9
1.2008.1 Available under license:
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:
http://www.opensource.org/licenses/mit-license.php

Copyright 2010 James Halliday (mail@substack.net)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2009 buffer 5.6.0 5.6.0
1.2009.1 Available under license :
   The MIT License (MIT)

   Copyright (c) Feross Aboukhadijeh, and other contributors.

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2010 express 4.13.4 4.13.4
1.2010.1 Available under license :
   (The MIT License)

   Copyright (c) 2009-2014 TJ Holowaychuk <tj@vision-media.ca>
   Copyright (c) 2013-2014 Roman Shylman <shylman+expressjs@gmail.com>
   Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. 
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY 
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, 
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE 
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2011 is-callable 1.2.4

1.2011.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all 
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE 
SOFTWARE.

1.2012 jboss-logging 3.1.1.GA
1.2012.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it.  You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price.  Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights.  These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you.  You must make sure that they, too, receive or can get the source
code.  If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it.  And you must show them these terms so they know their rights.
We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.
In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of
1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.
5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, 
but WITHOUT ANY WARRANTY; without even the implied warranty of 
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU 
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public 
License along with this library; if not, write to the Free Software 
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your 
school, if any, to sign a "copyright disclaimer" for the library, if 
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the 
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.2013 typedarray 0.0.6 0.0.6
1.2013.1 Available under license:

/*
 * Copyright (c) 2010, Linden Research, Inc.
 * Copyright (c) 2012, Joshua Bell
 * 
 * Permission is hereby granted, free of charge, to any person obtaining a copy 
 * of this software and associated documentation files (the "Software"), to deal 
 * in the Software without restriction, including without limitation the rights 
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
 * copies of the Software, and to permit persons to whom the Software is 
 * furnished to do so, subject to the following conditions:
 * 
 * The above copyright notice and this permission notice shall be included in 
 * all copies or substantial portions of the Software.
 * 
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR 
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, 
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE 
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER 
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
$LicenseInfo$
 */

// Original can be found at:
// https://bitbucket.org/lindenlab/llsd
// Modifications by Joshua Bell inexorabletash@gmail.com
// https://github.com/inexorabletash/polyfill

// ES3/ES5 implementation of the Khronos Typed Array Specification
// Ref: http://www.khronos.org/registry/typedarray/specs/latest/
// Date: 2011-02-01
// Variations:
// * Allows typed_array.get/set() as alias for subscripts (typed_array[])

1.2014 netty-transport-native-epoll
4.1.45.Final
1.2014.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Set the { @code TCP_MD5SIG} option on the socket. See { @code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
/**
 * Set the { @code TCP_QUICKACK} option on the socket. See http://linux.die.net/man/7/tcp"TCP_QUICKACK" for more details.
 */
/**

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollSocketChannelConfig.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Set the { @code TCP_MD5SIG} option on the socket. See { @code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollRecvByteAllocatorStreamingHandle.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollMode.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollDatagramChannelConfig.java
No license file was found, but licenses were detected in source scan.

/

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/AbstractEpollChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollDatagramChannel.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/NativeDatagramPacketArray.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollEventLoopGroup.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-jar/io/netty/channel/epoll/EpollTcpInfo.java

Open Source Used In DNAC Platform Arica 9233
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-
  jar/io/netty/channel/epoll/LinuxSocket.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-
  jar/io/netty/channel/epoll/NativeStaticallyReferencedJniMethods.java
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-
  jar/netty_epoll_linuxsocket.c
* /opt/ws_local/PERMITS_SQL/1074832144_1595959525.4/0/netty-transport-native-epoll-4.1.45-final-sources-
  jar/netty_epoll_linuxsocket.h

1.2015 batik-gvt 1.13

1.2015.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software's
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2016 ion-java 1.0.2

1.2016.1 Available under license:

Amazon Ion Java
Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2017 jctools-core 2.0.1
1.2017.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/SpscGrowableArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscGrowableArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpmcArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/MpscLinkedAtomicQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/MpscLinkedAtomicQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/LinkedQueueAtomicNode.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/LinkedQueueNode.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/ConcurrentCircularArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscCompoundQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/AtomicQueueFactory.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/util/UnsafeRefArrayAccess.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscUnboundedArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/BaseLinkedQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/package-info.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/NonBlockingIdentityHashMap.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/ConcurrentAutoTable.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/spec/Preference.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/BaseLinkedAtomicQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/SpscAtomicArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/SpmcArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/SpscArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/SpscLinkedQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscChunkedArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MpscLinkedQueue7.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/IndexedQueueSizeUtil.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/NonBlockingHashMap.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/SpscChunkedArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/MpscChunkedArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/NonBlockingHashSet.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/atomic/MpmcAtomicArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/atomic/SequencedAtomicReferenceArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/util/Pow2.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/QueueFactory.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/spec/Ordering.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/SpscUnboundedAtomicArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/SpscUnboundedArrayQueue.java
jar/org/jctools/queues/atomic/MpscAtomicArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/util/UnsafeAccess.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/BaseMpscLinkedArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/ConcurrentSequencedCircularArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/maps/NonBlockingHashMapLong.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/MessagePassingQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/atomic/SpmcAtomicArrayQueue.java
* /opt/cola/permits/1258286367_1643052498.42/0/jctools-core-2-0-1-sources-jar/org/jctools/queues/BaseSpscLinkedArrayQueue.java

1.2018 expat 2.1.0 7ubuntu0.16.04.5

1.2018.1 Available under license:
Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper
Copyright (c) 2001-2017 Expat maintainers

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2019 assertion-error 1.1.0

1.2019.1 Available under license:
No license file was found, but licenses were detected in source scan.

# AssertionError [[!Build Status](https://travis-ci.org/chaijs/assertion-error.png?branch=master)](https://travis-
ci.org/chaijs/assertion-error)

> Error constructor for test and validation frameworks that implements standardized AssertionError specification.

## Installation

### Node.js

`assertion-error` is available on [npm](http://npmjs.org).

$ npm install assertion-error

### Component

`assertion-error` is available as a [component](https://github.com/component/component).

$ component install chaijs/assertion-error

## License

(The MIT License)

Copyright (c) 2013 Jake Luer <jake@qualiancy.com> (http://qualiancy.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Found in path(s):
* /opt/cola/permits/1112206225_1607050024.11/0/chaijs-assertion-error-1-1-0-0-gfaa3f8c-1-tar-gz/chaijs-assertion-error-faa3f8c/README.md

No license file was found, but licenses were detected in source scan.
1.2020 guice-servlet 4.0
1.2020.1 Available under license:

Google Guice - Extensions - Servlet
Copyright 2006-2015 Google, Inc.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2021 semaphore-async-await 1.5.1 1.5.1

1.2021.1 Available under license:

Copyright (c) 2016 Jan Soendermann

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2022 optionator 0.8.3
1.2022.1 Available under license:
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.2023 es6-set 0.1.5
1.2023.1 Available under license:
Copyright (C) 2013 Mariusz Nowak (www.medikoo.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2024 guava 30.1.1-jre
1.2024.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution.”

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and eachContributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2025 lodash-max 4.0.1 4.0.1

1.2025.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2026 aws-java-sdk-s3 1.11.375

1.2026.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
   * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
   * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt for applicable license terms and NOTICE.txt for applicable notices.
   * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at http://aws.amazon.com/apache2.0
   * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either explicit or implied.
   */
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/internal/AbstractRepeatableCipherInputStream.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/DeleteVersionRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/package-info.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/SetBucketLoggingConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/SetBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/SetBucketVersioningConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/SetBucketTaggingConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/BucketVersioningConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/MultiFactorAuthentication.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/ObjectListing.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/GroupGrantee.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1.jar/com/amazonaws/services/s3/model/S3VersionSummary.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1.11-375-sources-1-
/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 *  http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */
/**
 * Interface for providing encryption materials.
 * Implementations are free to use any strategy for providing encryption
 * materials, such as simply providing static material that doesn't change,
 * or more complicated implementations, such as integrating with existing
 * key management systems.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/EncryptionMaterialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 *  http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing

* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/NotificationConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/PutInstructionFileRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/OnFileDelete.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SSECustomerKey.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/MaterialsDescriptionProvider.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SSEAwsKeyManagementParamsProvider.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/EncryptedGetObjectRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/MultipleFileTransferProgressUpdatingListener.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/S3ObjectIdBuilder.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/internal/S3SyncProgressListener.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/AmazonS3URI.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SSEAwsKeyManagementParams.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/PartCreationEvent.java
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/internal/crypto/MultipartUploadContext.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/GetBucketPolicyRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/AmazonS3ClientParams.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/GetBucketWebsiteConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/ListObjectsV2Result.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/GetObjectWebsiteConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/GetBucketLocationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/ListObjectsV2Request.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/TransferManagerBuilder.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/GetBucketLifecycleConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/DeleteBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/Tier.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/inventory/InventoryOptionalField.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketMetricsConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/analytics/AnalyticsFilterPredicate.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/S3CredentialsProviderChain.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/metrics/MetricsAndOperator.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/analytics/AnalyticsConfiguration.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/inventory/InventoryPredicateVisitor.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/OwnerOverride.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SetBucketAnalyticsConfigurationResult.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/metrics/MetricsPrefixPredicate.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/inventory/InventoryFrequency.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/lifecycle/LifecyclePredicateVisitor.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/analytics/AnalyticsS3BucketDestination.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysisDataExport.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/ListBucketInventoryConfigurationsResult.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/inventory/InventoryS3BucketDestination.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/analytics/AnalyticsFilter.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/inventory/InventoryPrefixPredicate.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysisDataExport.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/ListBucketInventoryConfigurationsResult.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/DeleteBucketAnalyticsConfigurationRequest.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/inventory/InventoryFormat.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/metrics/MetricsPredicateVisitor.java
*/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/metrics/MetricsConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketInventoryConfigurationResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/inventory/InventoryIncludedObjectVersions.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketMetricsConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/metrics/MetricsTagPredicate.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/metrics/MetricsNArOperation.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/analytics/AnalyticsNAryOperator.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/analytics/AnalyticsExportDestination.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/lifecycle/LifecycleTagPredicate.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/lifecycle/LifecycleFilterPredicate.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 */
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ExpressionType.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/Unmarshallers.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/ObjectTaggingProvider.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/InputSerialization.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/ObjectRestoreHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/S3ClientOptions.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/input/ObjectExpirationHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/CryptoConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/HeadBucketRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/AclXmlFactory.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/InitiateMultipartUploadRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/crypto/AdjustedRangeInputStream.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetRequestPaymentConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Region.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectsRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/AbstractHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/KMSEncryptionMaterials.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 */
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/crypto/SecuredCEK.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/crypto/KMSSecuredCEK.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance
 * with
 * the License. A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
 * permissions
 * and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/EncryptionConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ServerSideEncryptionByDefault.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ServerSideEncryptionRule.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/inventory/ServerSideEncryptionKMS.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketEncryptionResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/inventory/InventoryEncryption.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SseKmsEncryptedObjects.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/inventory/ServerSideEncryptionS3.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketEncryptionRequest.java
/*
 * Copyright 2011-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not
 * use this file except in compliance with the License. A copy of the License is
 * located at
 * 
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/waiters/AmazonS3Waiters.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * 
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/internal/CopyMonitor.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GenericBucketRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketCrossOriginConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketLifecycleConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ObjectExpirationResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/MultiObjectDeleteException.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/CORSRule.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectsResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ResponseHeaderOverrides.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ResponseHeaderHandlerChain.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/CopyPartResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectsResponse.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/MultiObjectDeleteResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/CopyPartRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/TransferMonitor.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketLifecycleConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ServerSideEncryptionResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketLifecycleConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ServerSideEncryptionResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ObjectExpirationResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetBucketNotificationConfigurationRequest.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*/
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

// The Amazon S3 DevPay token header is a special exception and can be safely moved

Found in path(s):
/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/AmazonS3Client.java

No license file was found, but licenses were detected in source scan.

Copyright (c) 2017. Amazon.com, Inc. or its affiliates. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

Found in path(s):
/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientBuilder.java
/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParams.java
/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/KeyFilter.java
/opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParamsWrapper.java

No license file was found, but licenses were detected in source scan.

Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/SkipMd5CheckStrategy.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/WebsiteConfiguration.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketAccelerateStatus.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/S3RequesterChargedHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/ObjectTaggingXmlFactory.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetObjectTaggingResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ObjectTagging.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
No license file was found, but licenses were detected in source scan.

/...
/*
 * Copyright 2015-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 *  http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/ServerSideEncryptionByDefaultStaxUnmarshaller.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteBucketReplicationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/QueueConfigurationStaxUnmarshaller.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/S3KeyFilterStaxUnmarshaller.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ReplicationDestinationConfig.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/request/S3HandlerContextKeys.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/S3VersionResultStaxUnmarshaller.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/S3AccelerateUnsupported.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Tag.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectAclRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectAclRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/DeleteObjectTaggingRequest.java
jar/com/amazonaws/services/s3/model/SetObjectAclRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/GetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/internal/CompleteMultipartUploadRetryCondition.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/ReplicationRule.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/BucketReplicationConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/NotificationConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/AbstractAmazonS3.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/HeadBucketResultHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/HeadBucketResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/FilterRule.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/LambdaConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/StatsStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/Filter.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/BucketNotificationConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/Exception/FileNotFoundException.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/TopicConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/GetBucketReplicationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/ServerSideEncryptionRuleStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/TopicConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/GetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/GetBucketEncryptionStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/FilterRuleStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/StatsStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/GetBucketEncryptionStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/LambdaConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/NotificationConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/GetObjectAclRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/ReplicationRuleStatus.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/HeadBucketResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/FilterRule.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/GetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/ServerSideEncryptionRuleStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/ServerSideEncryptionRuleStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/transform/TopicConfigurationStaxUnmarshall.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
jar/com/amazonaws/services/s3/model/GetBucketReplicationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
/*
 * Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * 
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * or in the "license" file accompanying this file. This file is distributed
 */

No license file was found, but licenses were detected in source scan.
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/RepeatableFileInputStream.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/RestUtils.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Grantee.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Owner.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/transform/XmlResponsesSaxParser.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketLoggingConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/AccessControlList.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/RepeatableInputStream.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/S3StringResponseHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/EmailAddressGrantee.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Bucket.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/Mimetypes.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/ServiceUtils.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/RequestPaymentConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/CanonicalGrantee.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketNotificationConfiguration.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/Grant.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/BucketTaggingConfiguration.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2017-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Header.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Utils.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Prelude.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/HeaderValue.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Message.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/HeaderType.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/eventstreaming/Checksums.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/SetObjectTaggingResponseHeaderHandler.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/GetObjectTaggingResponseHeaderHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/StaticEncryptionMaterialsProvider.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/Download.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/internal/DownloadImpl.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/iterable/S3Versions.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/iterable/S3Objects.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012-2018 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentEventStream.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/Progress.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/internal/eventstreaming/MessageDecoder.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/JSONOutput.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectRecordsInputStream.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/Stats.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/CompressionType.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentEventException.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/transform/SelectObjectContentEventUnmarshaller.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentEventVisitor.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentEvent.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/RequestProgress.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/JSONType.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/SelectObjectContentResult.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1.jar/com/amazonaws/services/s3/model/JSONInput.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 Amazon Technologies, Inc.
 */
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at:

http://aws.amazon.com/apache2.0

This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketTaggingConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/SetBucketReplicationConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketVersioningConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ListNextBatchOfObjectsRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketAccelerateConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/DownloadCallable.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ListNextBatchOfVersionsRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucket AccelerateConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/transfer/DownloadTaskImpl.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketLifecycleConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/ListNextBatchOfVersionsRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/GetBucketLoggingConfigurationRequest.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/internal/FileLocks.java

No license file was found, but licenses were detected in source scan.

/*

Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights
Reserved.

Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at

http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/internal/auth/S3SignerProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/transfer/internal/S3ProgressPublisher.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/transfer/internal/S3ProgressListenerChain.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/transfer/PauseException.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/transfer/PersistableDownload.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/transfer/exception/S3ObjectWrapper.java
Open Source Used In DNAC Platform Arica 9291
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

/**
 * Gets the optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin returning
 * results. Results are always ordered first lexicographically (i.e.
 * alphabetically) and then from most recent version to least recent
 * version.
 * <p>
 * A key marker must be specified when specifying a <code>versionIdMarker</code>.,
 * Results begin immediately after the version with the specified key and
 * version ID.
 * </p>
 *
 * @return The optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin
 * returning results.
 *
 * @see ListVersionsRequest#setVersionIdMarker(String)
 * @see ListVersionsRequest#withVersionIdMarker(String)
 */

Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-
  jar/com/amazonaws/services/s3/model/ListVersionsRequest.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 */
* S3 synchronous client for accessing Amazon S3.

* <h1>Appendix: Amazon S3 client-side encryption meta information</h1>

* <h2>Introduction</h2>

* This appendix summarizes the current crypto related meta information associated with an S3 object encrypted using SDK client-side encryption. In SDK's terminology, these meta information are classified as "user metadata".

* In general, the meta information is just a map of key/value pairs. Physically they are persisted either as metadata of the encrypted S3 object, or as a separate S3 object called an "instruction file" (which is just a JSON file in plaintext.) For all references to the Cipher Algorithm Names, Modes, and Padding, such as "AES/CBC/PKCS5Padding", please refer to Oracle's Java Cryptography Architecture Standard Algorithm Name Documentation for more details.

* <h2>S3 metadata vs instruction file</h2>

* Note when a key/value pair is stored as "user metadata" as part of the S3 metadata, the SDK always adds an additional prefix of "x-amz-meta-" to the key name. In contrast, the SDK doesn't add any prefix when the key names are stored in an instruction file. (The reason has to do with how the http headers are handled.)

* For example, "x-amz-key" would get stored as "x-amz-meta-x-amz-key" in S3 metadata. In contrast, "x-amz-key" would simply be stored as is (ie without the prefix) in an instruction file.

* All the keys described below are considered "user metadata" by the SDK.

* <h2>S3 v1 crypto meta information (supported by all SDK's)</h2>

* For non-KMS client side encryption, this v1 crypto meta information is used to support the "Encryption Only" crypto mode (which involves the use of AES/CBC/PKCS5Padding for content encryption, and AES/ECB for encrypting the one-time randomly generated data key). For KMS client-side encryption, the v2 crypto meta information is used.
<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-key</td>
<td>Content encrypting key (cek) in encrypted form, base64 encoded. The cek is randomly generated per S3 object, and is always an AES 256-bit key. The corresponding cipher is always “AES/CBC/PKCS5Padding”.</td>
</tr>
<tr>
<td>x-amz-iv</td>
<td>Randomly generated IV (per S3 object), base64 encoded</td>
</tr>
<tr>
<td>x-amz-matdesc</td>
<td>Customer provided material description in JSON (UTF8) format. Used to identify the client-side master key (ie used to encrypt/wrap the generated content encrypting key).</td>
</tr>
<tr>
<td>x-amz-unencrypted-content-length</td>
<td>Unencrypted content length (optional but should be specified whenever possible).</td>
</tr>
</tbody>
</table>

---

<h2>S3 v2 crypto meta information (currently supported by the Java SDK)</h2>

This v2 crypto meta information is used to support both authenticated encryption (which involves the use of `<code>AES/GCM/NoPadding</code>` for content encryption, and key wrapping for the one-time randomly generated data key), and KMS client-side encryption (which can either be encryption-only or authenticated encryption.)

<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-key-v2</td>
<td>CEK in key wrapped form. This is necessary so that the S3 encryption client that doesn't recognize the v2 format will not mistakenly decrypt S3 object encrypted in v2 format.</td>
</tr>
</tbody>
</table>
<table>
  <tr>
    <td><b>x-amz-iv</b></td>
    <td>Randomly generated IV (per S3 object), base64 encoded. (Same as v1.)</td>
  </tr>
  <tr>
    <td><b>x-amz-matdesc</b></td>
    <td>Customer provided material description in JSON format. (Same as v1). Used to identify the client-side master key.</td>
  </tr>
  <tr>
    <td><b>x-amz-unencrypted-content-length</b></td>
    <td>Unencrypted content length. (optional but should be specified whenever possible. Same as v1.)</td>
  </tr>
  <tr>
    <td><b>x-amz-wrap-alg</b></td>
    <td>Key wrapping algorithm used.<ul><li>Supported values: "AESWrap", "RSA/ECB/OAEPWithSHA-256AndMGF1Padding", "kms"</li></ul></td>
  </tr>
  <tr>
    <td><b>x-amz-cek-alg</b></td>
    <td>Content encryption algorithm used.<ul><li>Supported values: "AES/GCM/NoPadding", "AES/CBC/PKCS5Padding"</li></ul></td>
  </tr>
  <tr>
    <td><b>x-amz-tag-len</b></td>
    <td>Tag length (in bits) when AEAD is in use.<ul><li>Only applicable if AEAD is in use. This meta information is absent otherwise, or if KMS is in use.</li></ul></td>
  </tr>
</table>
KMS Integration for client-side encryption

All client-side KMS protected S3 objects are stored in v2 crypto meta information format. However, the key wrapping algorithm, \texttt{x-amz-wrap-alg}, is always set to \texttt{"kms"}.

The KMS Customer Master Key ID is currently stored as part of the material description, \texttt{x-amz-matdesc}, under the key-name \texttt{"kms_cmk_id"}.

Sample S3 crypto meta information for KMS client-side encryption

**CryptoMode: Encryption Only**

```
x-amz-meta-x-amz-key-v2:
CiAJ2GqwpJRnt4izujuwoalC2YdNRNSMeSlG5rw3pEI2J8hKnAQEBAwB4CdhsKSUZ7es7o8KGpQtmHTUTUjBLJRua8N6RCpiI1AAAB+MHwGCSqGSIb3DQEHBqBvMG0CAQAwaAYJKoZIhlvcNAQcBMB4GCWCGSAFlAwQBBljARBAYd0rzN3KHViWixDVcCARCAOhZ6xA8rob3Z0mNx3uKFieHKVyiIC/PIchALvGioVLnf86mcZJNpTXtwMkKkDocVj3Z1GjZTufFsOH
x-amz-iv: xqOHh+M6y0UogwG5SHJM3Q==
x-amz-unencrypted-content-length: 32768
x-amz-cek-alg: AES/CBC/PKCS5Padding
x-amz-wrap-alg: kms
x-amz-matdesc: \{"kms_cmk_id":\"d3fd2273-4ca0-4da5-b1c2-c89742ec6a26\"\}
```

**CryptoMode: Authenticated Encryption**

```
x-amz-key-v2:
CiAJ2GqwpJRnt4izujuwoalC2YdNRNSMeSlG5rw3pEI2J8hKnAQEBAwB4CdhsKSUZ7es7o8KGpQtmHTUTUjBLJRua8N6RCpiI1AAAB+MHwGCSqGSIb3DQEHBqBvMG0CAQAwaAYJKoZIhlvcNAQcBMB4GCWCGSAFlAwQBBljARBAYtFVvW4i6QmwcPK1ECARCAO1VzoM6Kxa9DHYNMm/BEbbjle2vLAL7IYf90kUoSqrdxgKxpNID9QaL/7928m1Djtld0bJpHefjm89C
x-amz-iv: VZBksyij6DhrUMye
x-amz-unencrypted-content-length: 32768
x-amz-cek-alg: AES/GCM/NoPadding
x-amz-wrap-alg: kms
x-amz-matdesc: \{"kms_cmk_id":\"d3fd2273-4ca0-4da5-b1c2-c89742ec6a26\"\}
x-amz-tag-len: 128
```

---

Open Source Used in DNAC Platform Arica 9296
* <h2>S3 metadata of an instruction file</h2>
* <p>The following key is always included as part of the S3 metadata of the
* instruction file (which is just an S3 object). Since the SDK treats this key
* as "user metadata", the key name is always prefixed with <b>x-amz-meta-</b>
* when physically stored.
* <p>
* <table border=1>
* <tr>
* <th>key</th>
* <th>description</th>
* </tr>
* <tr>
* <td><b>x-amz-crypto-instr-file</b></td>
* <td>the presence of this key is used to indicate this S3 object is an
* instruction file (of an associated client-side encrypted S3 object).</td>
* </tr>
* </table>
*
* No license file was found, but licenses were detected in source scan.
* <p>
* /*
* Copyright 2010-2018 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* *
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* *
* http://aws.amazon.com/apache2.0
* *
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
* *
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
* */
*
* No license file was found, but licenses were detected in source scan.
* Found in path(s):
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/package-info.java
* /opt/cola/permits/1180370208_1626285830.02/0/aws-java-sdk-s3-1-11-375-sources-1-jar/com/amazonaws/services/s3/model/S3Object.java
1.2027 liftoff 3.0.0 3.0.0

1.2027.1 Available under license:

Copyright (c) 2014 Tyler Kellen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2028 aircompressor 0.10

1.2028.1 Available under license:

Snappy Copyright Notices

=================================

* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011, Google Inc.<opensource@google.com>

Snappy License

==============

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2029 rimraf 2.6.3
1.2029.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2030 commons-validator 1.7

1.2030.1 Available under license:
Apache Commons Validator
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
The Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2031 mime-types 2.1.19 2.1.19

1.2031.1 Available under license :

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2032 is-npm 5.0.0

1.2032.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2033 connect 3.4.1 3.4.1

1.2033.1 Available under license:
(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2034 calcite-core 1.21.0-drill-r2

1.2034.1 Available under license:

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to you under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# Database of contributors to Apache Calcite.
# Pages such as developer.md use this data.
# List must be sorted by first name, last name.
#
# name: Alan Gates
# emeritus: 2018/05/04
# apacheId: gates
githubId: alanfgates
org: Hortonworks
role: PMC
- name: Aman Sinha
  apacheId: amansinha
  githubId: amansinha100
  org: MapR
  role: PMC
- name: Andrei Sereda
  apacheId: sereda
  githubId: asereda-gs
  role: Committer
- name: Ashutosh Chauhan
  apacheId: hashutosh
  githubId: ashutoshc
  org: Hortonworks
  role: PMC
- name: Chris Baynes
  apacheId: cbaynes
  githubId: chris-baynes
  org: Contiamo
  role: Committer
- name: Christian Beikov
  apacheId: cbeikov
  githubId: beikov
  org: Blazebit
  role: Committer
- name: Chunwei Lei
  apacheId: chunwei
  githubId: chunweilei
  org: Alibaba
  role: Committer
- name: Danny Chan
  apacheId: danny0405
  githubId: danny0405
  org: Alibaba
  role: Committer
- name: Edmon Begoli
  apacheId: ebegoli
  githubId: ebegoli
  org: Oak Ridge National Laboratory
  role: Committer
- name: Francis Chuang
  apacheId: francischuang
  githubId: F21
  org: Boostport
  role: PMC Chair
- name: Gian Merlino
  apacheId: gian
  githubId: gianm
  org: Imply
  role: Committer
- name: Haisheng Yuan
  apacheId: hyuan
  githubId: hsyuan
  org: Alibaba
  role: Committer
- name: Hongze Zhang
  apacheId: hongze
  githubId: zhztheplayer
  org: Tencent
  role: Committer
- name: James R. Taylor
  apacheId: jamesTaylor
  githubId: JamesRTaylor
  org: Salesforce
  role: PMC
- name: Jacques Nadeau
  apacheId: jacques
  githubId: jacques-n
  org: Dremio
  role: PMC
- name: Jess Balint
  apacheId: jbalint
  githubId: jbalint
  org: Stardog
  role: Committer
- name: Jess Camacho Rodriguez
  apacheId: jcamacho
  githubId: jcamachor
  org: Hortonworks
  role: PMC
- name: Jinfeng Ni
  apacheId: jni
  githubId: jinfengni
  org: MapR
  role: PMC
- name: John Pullokkaran
  apacheId: jpullokk
  githubId: jpullokkaran
  org: Hortonworks
  role: PMC
- name: Josh Elser
  apacheId: elserj
  githubId: joshelser
org: Hortonworks
dc: PMC
- name: Julian Hyde
apacheId: jhyde
githubId: julianhyde
org: Looker
dc: PMC
homepage: http://people.apache.org/~jhyde
- name: Kevin Liew
apacheId: kliew
githubId: kliewkliw
org:
dc: Committer
- name: Kevin Risden
apacheId: krisden
githubId: risdenk
org:
dc: PMC
- name: Laurent Goujon
apacheId: laurent
githubId: laurentgo
org: Dremio
dc: PMC
- name: Maryann Xue
apacheId: maryannxue
githubId: maryannxue
org: Intel
dc: PMC
- name: Michael Mior
apacheId: mmior
githubId: michaelmior
org: Rochester Institute of Technology
dc: PMC
homepage: https://michael.mior.ca/
- name: Milinda Pathirage
apacheId: milinda
githubId: milinda
org: Indiana University
dc: Committer
homepage: http://milinda.pathirage.org/
- name: MinJi Kim
apacheId: minji
githubId: minji-kim
org: Oracle
dc: Committer
- name: Nick Dimiduk
apacheId: ndimiduk
githubId: ndimiduk
- name: Nishant Bangarwa
  apacheId: nishant
  githubId: nishantmonu51
  org: Hortonworks
  role: Committer
- name: Ruben Quesada Lopez
  apacheId: rubenql
  githubId: rubenada
  org: TIBCO
  role: Committer
- name: Sergey Nuyanzin
  apacheId: snuyanzin
  githubId: snuyanzin
  org: EPAM
  role: Committer
- name: Shuyi Chen
  apacheId: shuyichen
  githubId: suez1224
  org: Uber
  role: Committer
- name: Slim Bouguerra
  apacheId: bslim
  githubId: b-slim
  org: Hortonworks
  role: Committer
- name: Stamatis Zampetakis
  apacheId: zabetak
  githubId: zabetak
  org: TIBCO
  role: PMC
- name: Steven Noels
  apacheId: stevenn
  githubId: stevenn
  org: NGData
  role: PMC
- name: Ted Dunning
  apacheId: tdunning
  githubId: tdunning
  org: MapR
  role: PMC
  avatar: https://mapr.com/blog/author/ted-dunning/assets/tdunning-panama.jpg
- name: Vladimir Sitnikov
  apacheId: vladimirsitnikov
  githubId: vlsi
  org: NetCracker
role: PMC
- name: Volodymyr Vysotskyi
  apacheId: volodymyr
  githubId: vvsotskyi
  org:
  role: PMC
- name: Zhen Wang
  apacheId: zhenw
  githubId: zinking
  org:
  role: Committer
- name: Zhiqiang He
  apacheId: zhiqianghe
  githubId: Zhiqiang-He
  org: Huawei
  role: Committer
- name: Zhiwei Peng
  apacheId: zhiwei
  githubId: pengzhiwei2018
  org: Ant Financial Group
  role: Committer
- name: Zoltan Haindrich
  apacheId: kgyrtkirk
  githubId: kgyrtkirk
  org: Hortonworks
  role: Committer
# End contributors.yml

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-----------------------------------------------------------------------

APACHE CALCITE SUBCOMPONENTS:

The Apache Calcite project contains subcomponents with separate copyright
notices and license terms. Your use of the source code for the these
subcomponents is subject to the terms and conditions of the following licenses.

-----------------------------------------------------------------------
The MIT License
-----------------------------------------------------------------------

The Apache Calcite project bundles the following files under the MIT License:

- site
  Parts of the web site generated by Jekyll (http://jekyllrb.com/)
  Copyright (c) 2008-2015 Tom Preston-Werner
- site/_sass/_font-awesome.scss
  Font-awesome css files v4.1.0 (http://fortawesome.github.io/Font-Awesome/)
  Copyright (c) 2013 Dave Gandy
- site/_sass/_normalize.scss
  normalize.css v3.0.2 | git.io/normalize
  Copyright (c) Nicolas Gallagher and Jonathan Neal
- site/_sass/_gridism.scss
  Gridism: A simple, responsive, and handy CSS grid by @cobyism
  https://github.com/cobyism/gridism
  Copyright (c) 2013 Coby Chapple
- site/js/html5shiv.min.js
  HTML5 Shiv 3.7.2 | @afarkas @dalton @jon_neal @rem
- site/js/respond.min.js
  Respond.js v1.4.2: min/max-width media query polyfill
  Copyright 2013 Scott Jehl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------
The Open Font License
The Apache Calcite project bundles the following fonts under the
SIL Open Font License (OFL) - http://scripts.sil.org/OFL/
- site/fonts/fontawesome-webfont.*
  Font-awesome font files v4.0.3 (http://fortawesome.github.io/Font-Awesome/)
Apache Calcite
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.

The web site includes files generated by Jekyll.

1.2035 yauzl 2.10.0 2.10.0
1.2035.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Josh Wolfe

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
1.2036 ant-launcher 1.10.9

1.2036.1 Available under license:

/*
 * Apache License
 * Version 2.0, January 2004
 * http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction,
 * and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by
 * the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all
 * other entities that control, are controlled by, or are under common
 * control with that entity. For the purposes of this definition,
 * "control" means (i) the power, direct or indirect, to cause the
 * direction or management of such entity, whether by contract or
 * otherwise, or (ii) ownership of fifty percent (50%) or more of the
 * outstanding shares, or (iii) beneficial ownership of such entity.
 *
 * "You" (or "Your") shall mean an individual or Legal Entity
 * exercising permissions granted by this License.
 *
 * "Source" form shall mean the preferred form for making modifications,
 * including but not limited to software source code, documentation
 * source, and configuration files.
 *
 * "Object" form shall mean any form resulting from mechanical
 * transformation or translation of a Source form, including but
 * not limited to compiled object code, generated documentation,
 * and conversions to other media types.
 *
 * "Work" shall mean the work of authorship, whether in Source or
 * Object form, made available under the License, as indicated by a
 * copyright notice that is included in or attached to the work
 * (an example is provided in the Appendix below).
 *
 * "Derivative Works" shall mean any work, whether in Source or Object
 * form, that is based on (or derived from) the Work and for which the
 * editorial revisions, annotations, elaborations, or other modifications
 * represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.
The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

____________________________________

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
Apache Ant
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.
1.2037 log-symbols 3.0.0

1.2037.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2038 swagger-core 2.0.7

1.2038.1 Available under license:
Copyright 2018 SmartBear Software

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2039 js-cookie 2.1.1

1.2039.1 Available under license:
Copyright 2014 Klaus Hartl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2040 colord 1.2.12-1ubuntu1

1.2040.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c': they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation’s software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author’s protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

&lt;one line to give the program's name and a brief idea of what it does.&gt;
Copyright (C) &lt;year&gt; &lt;name of author&gt;

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 
`Gnomovision' (which makes passes at compilers) written by James Hacker.

&lt;signature of Ty Coon&gt;, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.2041 is-unc-path 1.0.0 1.0.0

1.2041.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2042 object-copy 0.1.0 0.1.0
Available under license:
The MIT License (MIT)

Copyright (c) 2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Available under license:
(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
1.2044 xmpcore 6.0.6
1.2044.1 Available under license:
ADOBE CONFIDENTIAL

Copyright 2011-2016 Adobe Systems Incorporated
All Rights Reserved.

NOTICE: All information contained herein is, and remains
the property of Adobe Systems Incorporated and its suppliers,
if any. The intellectual and technical concepts contained
herein are proprietary to Adobe Systems Incorporated and its
suppliers and may be covered by U.S. and Foreign Patents,
patents in process, and are protected by trade secret or copyright law.
Dissemination of this information or reproduction of this material
is strictly forbidden unless prior written permission is obtained
from Adobe Systems Incorporated.

1.2045 locate-path 3.0.0
1.2045.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2046 jersey-server 2.25.1
1.2046.1 Available under license:
   CDDL+GPL License
   http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.2047 is-my-ip-valid 1.0.0
1.2047.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2048 camelcase 6.2.1
1.2048.1 Available under license:
   MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2051 guice-assistedinject 3.0
1.2051.1 Available under license:

Google Guice - Extensions - AssistedInject
Copyright 2006-2011 Google, Inc.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2052 oas-kit-common 1.0.8 1.0.8

1.2052.1 Available under license:

BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2053 liquibase 3.8.7

1.2053.1 Available under license:

SOFTWARE LICENSE SUBSCRIPTION AND SUPPORT AGREEMENT

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THIS SERVICE.

BY USING THE SOFTWARE OR CLICKING AGREE, CUSTOMER IS AGREEING TO BE BOUND BY THIS AGREEMENT. IF YOU ARE AGREEING TO THIS AGREEMENT ON BEHALF OF OR FOR THE BENEFIT OF YOUR EMPLOYER OR A THIRD PARTY, THEN YOU REPRESENT AND WARRANT THAT YOU HAVE THE NECESSARY AUTHORITY TO AGREE TO THIS AGREEMENT ON YOUR EMPLOYER'S OR THAT THIRD PARTY'S BEHALF.

This agreement is between Datical, Inc. a Delaware corporation (Datical) and you, the customer entering into this agreement (Customer). The LiquibasePro software, updates, documentation and license keys provided to Customer (Software) are licensed and are not sold.

SCOPE. This agreement describes the licensing of the Software and support provided to Customer under an order. Third party contractors and majority owned affiliates of Customer can use and access the Software under the terms of this agreement, and Customer is responsible for each of their compliance with such terms.

1. LICENSE. Subject to the other terms of this agreement, Datical grants Customer, under an order, a term based non-exclusive, non-transferable license up to the license capacity purchased to use the Software only in Customers internal business operations. Third party licenses delivered with the Software
may be subject to additional license rights and restrictions as set forth at
https://download.liquibase.org/lbpro-third-party-licenses/.

2. RESTRICTIONS. Customer may not:
   a. Transfer, assign, sublicense, rent the Software, create derivative works of
      the Software, or use it in any type of service provider environment;
   b. Reverse engineer, decompile, disassemble, or translate the Software; or
   c. Evaluate the Software for the purpose of competing with Liquibase Pro or
      operate the Software other than in accordance with its technical documentation.

3. PROPRIETARY RIGHTS AND MUTUAL CONFIDENTIALITY.
   a. Proprietary Rights. The Software, workflow processes, user interface,
      designs and other technologies provided by Datical as part of the Software are
      the proprietary property of Datical and its licensors, and all right, title and
      interest in and to such items, including all associated intellectual property
      rights, remain only with Datical and its licensors. The Software is protected
      by copyright and other intellectual property laws. Customer may not remove any
      product identification, copyright, trademark or other notice from the Software.
      Datical reserves all rights not expressly granted.
   b. Mutual Confidentiality. Recipient may not disclose Confidential Information
      of Discloser to any third party or use the Confidential Information in
      violation of this agreement.
      Confidential Information means all information that is disclosed to the
      recipient (Recipient) by the discloser (Discloser), and includes, among other
      things:
      * any and all information relating to products or services provided by a
        Discloser, software code, flow charts, techniques, specifications, development
        and marketing plans, strategies, and forecasts;
      * as to Datical, the Software and the terms of this agreement, including
        without limitation, all pricing information.
      Confidential Information excludes information that:
      * was rightfully in Recipient's possession without any obligation of
        confidentiality before receipt from the Discloser;
      * is or becomes a matter of public knowledge through no fault of Recipient;
      * is rightfully received by Recipient from a third party without violation of
        a duty of confidentiality; or
      * is independently developed by or for Recipient without use or access to the
        Confidential Information.
      Recipient may disclose Confidential Information if required by law, but it will
      attempt to provide notice to the Discloser in advance so it may seek a
      protective order. Each party acknowledges that any misuse of the other party's
      Confidential Information may cause irreparable harm for which there is no
      adequate remedy at law. Either party may seek immediate injunctive relief in
      such event.

4. WARRANTY, REMEDY and DISCLAIMER. For new license purchases, Datical warrants
   that the Software will perform in substantial accordance with its accompanying
This warranty will not apply to any problems caused by software not licensed to Customer by Datical, use other than in accordance with the technical documentation, or misuse of the Software. The warranty only covers problems reported to Datical during the warranty period or 30 days after. Customer will cooperate with Datical in resolving any warranty claim.

A. EXCLUSIVE REMEDY AND SOLE LIABILITY. Datical will use commercially reasonable efforts to remedy covered warranty claims within a reasonable period of time or replace the Software, or if Datical cannot do so it will refund to Customer the license fee paid. THIS REMEDY IS CUSTOMERS EXCLUSIVE REMEDY, AND DATICALS SOLE LIABILITY FOR THESE WARRANTY CLAIMS.

B. DISCLAIMER OF WARRANTIES. DATICAL DISCLAIMS ALL OTHER EXPRESS AND IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER UNDERSTANDS THAT THE SOFTWARE MAY NOT BE ERROR FREE, AND USE MAY BE INTERRUPTED.

5. TERMINATION. This agreement expires at the end of the license period specified. Either party may terminate this agreement upon a material breach of the other party after a 30 days notice/cure period, if the breach is not cured during such time period. Upon terminationof this agreement or a license, Customer must discontinue using the Software, de-install and destroy or return the Software and all copies, within 5 days. Upon Datical's request, Customer will provide written certification of such compliance.

6. SUPPORT. Daticals technical support and maintenance services (Support) is included with the license fees. Customer will receive best efforts e-mail response for support within two business days from receipt of issue. Datical may change its Support terms, but Support will not materially degrade during any Support term.

7. LIMIT ON LIABILITY. There may be situations in which (as a result of material breach or other liability) Customer is entitled to make a claim against Datical. In each situation (regardless of the form of the legal action (e.g. contract or tort claims)), Datical is not responsible for any damage and does not have any liability beyond the greater of the amount paid or payable by Customer to Datical within the 12 months prior to the event that gave rise to the claim and in no event is Datical responsible for any: loss of data or information; lost profits, revenue, or productivity; or other special, consequential, incidental or indirect damages.

8. INTELLECTUAL PROPERTY INDEMNITY

a. Defense of Third Party Claims. Datical will defend or settle any third party claim against Customer to the extent that such claim alleges that the Software violates a copyright, patent, trademark or other intellectual property right, if Customer, promptly notifies Datical of the claim in writing, cooperates with Datical in the defense, and allows Datical to solely control the defense or settlement of the claim. Costs. Datical will pay infringement claim defense
costs incurred as part of its obligations above, and Datical negotiated settlement amounts, and court awarded damages. Process. If such a claim appears likely, then Datical may modify the Software, procure the necessary rights, or replace it with the functional equivalent. If Datical determines that none of these are reasonably available, then Datical may terminate the Software and refund any prepaid and unused fees. Exclusions. Datical has no obligation for any claim arising from: Datical’s compliance with Customer’s specifications; A combination of the Software with other technology where the infringement would not occur but for the combination; or Technology not provided by Datical. THIS SECTION CONTAINS CUSTOMERS EXCLUSIVE REMEDIES AND DATICAL’S SOLE LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT CLAIMS.

9. GOVERNING LAW AND EXCLUSIVE FORUM. This agreement is governed by the laws of the State of Delaware (without regard to conflicts of law principles) for any dispute between the parties or relating in any way to the subject matter of this agreement. Any suit or legal proceeding must be exclusively brought in the federal or state courts for Travis County, Texas, and Customer submits to this personal jurisdiction and venue. Nothing in this agreement prevents either party from seeking injunctive relief in a court of competent jurisdiction. The prevailing party in litigation is entitled to recover its attorneys fees and costs from the other party.

10. OTHER TERMS.
   a. Entire Agreement. This agreement and the order constitute the entire agreement between the parties and supersede any prior or contemporaneous negotiations or agreements, whether oral or written, related to this subject matter. Customer is not relying on any representation concerning this subject matter, oral or written, not included in this agreement. No representation, promise or inducement not included in this agreement is binding.
   b. Non-Assignment. Neither party may assign or transfer this agreement to a third party, nor delegate any duty, except that the agreement and all orders may be assigned, without the consent of the other party, as part of a merger, or sale of all or substantially all of the business or assets, of a party.
   c. Independent Contractors. The parties are independent contractors with respect to each other.
   d. Enforceability. If any term of this agreement is invalid or unenforceable, the other terms remain in effect.
   e. Survival of Terms and Force Majeure. All terms that by their nature survive termination of this agreement for each party to receive the benefits and protections of this agreement, will survive. Neither party is liable for events beyond its reasonable control, including, without limitation force majeure events.
   f. Modification Only in Writing. No modification or waiver of any term of this agreement is effective unless signed by both parties.
   g. Export Compliance. Customer must comply with all applicable export control laws of the United States, foreign jurisdictions and other applicable laws and regulations.
   h. US GOVERNMENT RESTRICTED RIGHTS. If Customer is a United States government
agency or acquired the license to the Software hereunder pursuant to a
government contract or with government funds, then as defined in FAR 2.101,
DFAR 252.227-7014(a)(1) and DFAR 252.227-7014(a)(5) or otherwise, all
Software provided in connection with this agreement are commercial items,
commercial computer software or commercial computer software documentation.
Consistent with DFAR 227.7202 and FAR 12.212, any use, modification,
reproduction, release, performance, display, disclosure or distribution by or
for the United States government is governed solely by the terms of this
agreement and is prohibited except to the extent permitted by the terms of this
agreement.

i. No PO Terms. Datical rejects conflicting terms of a Customers
form-purchasing document.

j. Order of Precedence. If there is an inconsistency between this agreement and
an order, the order prevails.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS
"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are
not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the
Program.
Each Contributor must identify itself as the originator of its Contribution, if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore, if
a Contributor includes the Program in a commercial product offering, such
Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify
every other Contributor (“Indemnified Contributor”) against any losses, damages
and costs (collectively "Losses") arising from claims, lawsuits and other legal
actions brought by a third party against the Indemnified Contributor to the
extent caused by the acts or omissions of such Commercial Contributor in
connection with its distribution of the Program in a commercial product
offering. The obligations in this section do not apply to any claims or Losses
relating to any actual or alleged intellectual property infringement. In order
to qualify, an Indemnified Contributor must: a) promptly notify the Commercial
Contributor in writing of such claim, and b) allow the Commercial Contributor
to control, and cooperate with the Commercial Contributor in, the defense and
any related settlement negotiations. The Indemnified Contributor may
participate in any such claim at its own expense.
For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If that
Commercial Contributor then makes performance claims, or offers warranties
related to Product X, those performance claims and warranties are such
Commercial Contributor's responsibility alone. Under this section, the
Commercial Contributor would have to defend claims against the other
Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial
Contributor must pay those damages.

5. NO WARRANTY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using
and distributing the Program and assumes all risks associated with its exercise
of rights under this Agreement, including but not limited to the risks and
costs of program errors, compliance with applicable laws, damage to or loss of
data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the
intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year
after the cause of action arose. Each party waives its rights to a jury trial
in any resulting litigation.

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license
document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2
because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share
and change it. By contrast, the GNU General Public Licenses are intended to
guarantee your freedom to share and change free software--to make sure the
software is free for all its users.

This license, the Library General Public License, applies to some specially
designated Free Software Foundation software, and to any other libraries whose
authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our
General Public Licenses are designed to make sure that you have the freedom to
distribute copies of free software (and charge for this service if you wish),
that you receive source code or can get it if you want it, that you can change
the software or use pieces of it in new free programs; and that you know you
can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny
you these rights or to ask you to surrender the rights. These restrictions
translate to certain responsibilities for you if you distribute copies of the
library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a
fee, you must give the recipients all the rights that we gave you. You must
make sure that they, too, receive or can get the source code. If you link a
program with the library, you must provide complete object files to the
recipients so that they can relink them with the library, after making changes
to the library and recompiling it. And you must show them these terms so they
know their rights.
Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the
library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language.

(Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications
or work under the terms of Section 1 above, provided that you also meet all of
these conditions:

a) The modified work must itself be a software library.
b) You must cause the files modified to carry prominent notices stating that
you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all
third parties under the terms of this License.
d) If a facility in the modified Library refers to a function or a table of
data to be supplied by an application program that uses the facility, other
than as an argument passed when the facility is invoked, then you must make a
good faith effort to ensure that, in the event an application does not supply
such function or table, the facility still operates, and performs whatever part
of its purpose remains meaningful.
(For example, a function in a library to compute square roots has a purpose
that is entirely well-defined independent of the application. Therefore,
Subsection 2d requires that any application-supplied function or table used by
this function must be optional: if the application does not supply it, the
square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable
sections of that work are not derived from the Library, and can be reasonably
considered independent and separate works in themselves, then this License, and
its terms, do not apply to those sections when you distribute them as separate
works. But when you distribute the same sections as part of a whole which is a
work based on the Library, the distribution of the whole must be on the terms
of this License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your
rights to work written entirely by you; rather, the intent is to exercise the
right to control the distribution of derivative or collective works based on
the Library.

In addition, mere aggregation of another work not based on the Library with the
Library (or with a work based on the Library) on a volume of a storage or
distribution medium does not bring the other work under the scope of this
License.

3. You may opt to apply the terms of the ordinary GNU General Public License
instead of this License to a given copy of the Library. To do this, you must
alter all the notices that refer to this License, so that they refer to the
ordinary GNU General Public License, version 2, instead of to this License. (If
a newer version than version 2 of the ordinary GNU General Public License has
appeared, then you can specify that version instead if you wish.) Do not make
any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so
the ordinary GNU General Public License applies to all subsequent copies and
derivative works made from that copy.
This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange. If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License. However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and
reverse engineering for debugging such modifications.
You must give prominent notice with each copy of the work that the Library is
used in it and that the Library and its use are covered by this License. You
must supply a copy of this License. If the work during execution displays
copyright notices, you must include the copyright notice for the Library among
them, as well as a reference directing the user to the copy of this License.
Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source
code for the Library including whatever changes were used in the work (which
must be distributed under Sections 1 and 2 above); and, if the work is an
executable linked with the Library, with the complete machine-readable "work
that uses the Library", as object code and/or source code, so that the user can
modify the Library and then relink to produce a modified executable containing
the modified Library. (It is understood that the user who changes the contents
of definitions files in the Library will not necessarily be able to recompile
the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to
give the same user the materials specified in Subsection 6a, above, for a
charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a
designated place, offer equivalent access to copy the above specified materials
from the same place.

d) Verify that the user has already received a copy of these materials or that
you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must
include any data and utility programs needed for reproducing the executable
from it. However, as a special exception, the source code distributed need not
include anything that is normally distributed (in either source or binary form)
with the major components (compiler, kernel, and so on) of the operating system
on which the executable runs, unless that component itself accompanies the
executable.

It may happen that this requirement contradicts the license restrictions of
other proprietary libraries that do not normally accompany the operating
system. Such a contradiction means you cannot use both them and the Library
together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library
side-by-side in a single library together with other library facilities not
covered by this License, and distribute such a combined library, provided that
the separate distribution of the work based on the Library and of the other
library facilities is otherwise permitted, and provided that you do these two
things:

a) Accompany the combined library with a copy of the same work based on the
Library, uncombined with any other library facilities. This must be distributed
under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it
is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries
If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.

Copyright (C) year name of author

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

signature of Ty Coon, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

MIT License

Copyright (c) <year> <copyright holders>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Open Source Used In DNAC Platform Arica 9372
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2054 @babel/code-frame 7.10.4
1.2054.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANDABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2055 @babel/generator 7.11.5
1.2055.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2056 kotlin 1.4.10
1.2056.1 Available under license :
No license file was found, but licenses were detected in source scan.

"version":3,"file":"kotlin.js","sources":]
"wrapper.js","arrayUtils.js","callableReferenceUtils.js","conversions.js","core.js","long.js","markerFunctions.js","misc.js","polyfills.js","rtti.js","runtime/arrayUtils.kt","runtime/Enum.kt","primitiveCompanionObjects.kt","generated/_Arrays.kt","generated/_Ranges.kt","kotlin/UByte.kt","kotlin/UInt.kt","kotlin/UShort.kt","Ranges.kt","collections/Collections.kt","collections/Maps.kt","collections/Sets.kt","text/StringNumberConversions.kt","kotlin/UnsignedUtils.kt","collections/Iterables.kt","collections/Sequences.kt","util/Preconditions.kt","generated/_ArraysJs.kt","comparisons/Comparisons.kt","util/Standard.kt","generated/_ComparisonsJs.kt","kotlin/ULong.kt","generated/_Collections.kt","kotlin/collections.kt","collections/Iterators.kt","generated/_Comparisons.kt","generated/_Maps.kt","generated/_Sequences.kt","generated/_Sets.kt","generated/_Strings.kt","text/Strings.kt","kotlin/text/string.kt","kotlin/UByteArray.kt","kotlin/UIntArray.kt","kotlin/ULongArray.kt","kotlin/UShortArray.kt","
Kotlin.toByte(Kotlin.numberToInInt(a));

Kotlin.toChar(Kotlin.numberToInInt(a));

Kotlin.toDouble = function (a) {
    return

Kotlin.toChar(Kotlin.numberToInInt(a));

Kotlin.hashCode = function (obj) {
    if (obj == null) {
        return 0;
    }
    else if (Kotlin.isArrayish(obj)) {
        return "[...]";
    }
    else {
        return obj.toString();
    }
}

Kotlin.toString = function (o) {
    if (o == null) {
        return "null";
    }
    else if (Kotlin.isArrayish(o)) {
        return "[...]";
    }
    else {
        return o.toString();
    }
}

Kotlin.hashCode = function (obj) {
    if (obj == null) {
        return 0;
    }
    else if (Kotlin.isArrayish(obj)) {
        return "[...]";
    }
    else {
        return obj.toString();
    }
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    if (typeof obj1 === "object") {
        return obj1.equals(obj2);
    }
    if (typeof obj2 === "object") {
        return obj2.equals(obj1);
    }
    var objType = typeof obj1;
    if ("function" === objType) {
        return obj1(obj2);
    }
    if ("number" === objType) {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2);
    }
    var str = String(obj);
    return obj.toString();
}
Open Source Used In DNAC Platform Arica 9380

```javascript
{
    kind: "class",
    simpleName: "Long",
    interfaces:
```}

// NOTE: Common constant values

/**
 * A cache of the Long representations of small integer values.
 * @type {!Object}
 * @private
 */
Kotlin.Long.IntCache_ = {};

/**
 * Returns a Long representing the given (32-bit) integer value.
 * @param {number} number The number in question.
 * @return {!Kotlin.Long} The corresponding Long value.
 */
Kotlin.Long.fromInt = function(value) {
  if (-128 <= value && value < 128) {
    var cachedObj = Kotlin.Long.IntCache_[value];
    if (cachedObj) {
      return cachedObj;
    }
  }

  var obj = new Kotlin.Long(value | 0, value < 0 ? -1 : 0);
  if (-128 <= value && value < 128) {
  }
  return obj;
};

/**
 * Returns a Long representation of the given string, written using the given
 * radix in which the text is written.
 * @param {string} str The textual representation of the Long.
 * @param {number=} opt_radix The radix in which the text is written.
 * @return {!Kotlin.Long} The corresponding Long value.
 */
Kotlin.Long.fromString = function(str, opt_radix) {
  if (str.length == 0) {
    throw Error('number format error: empty string');
  }

  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  if (str.charAt(0) == '-') {
    return Kotlin.Long.fromString(str.substring(1), radix).negate();
  } else if (str.indexOf('-') >= 0) {
    throw Error('number format error: interior "-" character: ' + str);
  }

  // Do several (8) digits each time through the loop, so as to minimize
  // the calls to the very expensive emulated div.
  var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 8));

  var result = Kotlin.Long.ZERO;
  for (var i = 0; i < str.length; i += 8) {
    var size = Math.min(8, str.length - i);
    var value = parseInt(str.substring(i, i + size), radix);
    if (size < 8) {
      var power = Kotlin.Long.fromNumber(Math.pow(radix, size));
      result = result.multiply(power).add(Kotlin.Long.fromNumber(value));
    } else {
      result = result.multiply(radixToPower);
      result = result.add(Kotlin.Long.fromNumber(value));
    }
  }
  return result;
};

// NOTE: the compiler should inline these constant values below and then remove them
// from the code. These variables, so there should be no runtime penalty for these:

/** @type {number} */
Kotlin.Long.ZERO = Kotlin.Long.fromInt(0);

/** @type {!Kotlin.Long} */
```
*/\nKotlin.Long.ONE = Kotlin.Long.fromInt(1);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.NEG_ONE =
Kotlin.Long.fromInt(-1);\n\n\n/** @type {!Kotlin.Long} */\nKotlin.Long.MAX_VALUE =\n
Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);\n\n\n/** @type {!Kotlin.Long}
*/\nKotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);\n\n\n/**\n * @type {!Kotlin.Long}\n *
@private\n */\nKotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);\n\n\n/** @return {number} The
value, assuming it is a 32-bit integer. */\nKotlin.Long.prototype.toInt = function() {\n return this.low_;\n};\n\n\n/**
@return {number} The closest floating-point representation to this value. */\nKotlin.Long.prototype.toNumber =
function() {\n return this.high_ * Kotlin.Long.TWO_PWR_32_DBL_ +\n
this.getLowBitsUnsigned();\n};\n\n/** @return {number} The 32-bit hashCode of this value.
*/\nKotlin.Long.prototype.hashCode = function() {\n return this.high_ ^ this.low_;\n};\n\n/**\n * @param
{number=} opt_radix The radix in which the text should be written.\n * @return {string} The textual representation
of this value.\n * @override\n */\nKotlin.Long.prototype.toString = function(opt_radix) {\n var radix = opt_radix ||
10;\n if (radix < 2 || 36 < radix) {\n throw Error('radix out of range: ' + radix);\n }\n\n if (this.isZero()) {\n
return '0';\n }\n\n if (this.isNegative()) {\n if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n
// We need to
change the Long value before it can be negated, so we remove\n
// the bottom-most digit in this base and then
recurse to do the rest.\n
var radixLong = Kotlin.Long.fromNumber(radix);\n
var div = this.div(radixLong);\n
var rem = div.multiply(radixLong).subtract(this);\n
return div.toString(radix) + rem.toInt().toString(radix);\n }
else {\n
return '-' + this.negate().toString(radix);\n }\n }\n\n // Do several (6) digits each time through the loop,
so as to\n // minimize the calls to the very expensive emulated div.\n var radixToPower =
Kotlin.Long.fromNumber(Math.pow(radix, 6));\n\n var rem = this;\n var result = '';\n while (true) {\n var
remDiv = rem.div(radixToPower);\n var intval = rem.subtract(remDiv.multiply(radixToPower)).toInt();\n var
digits = intval.toString(radix);\n\n rem = remDiv;\n if (rem.isZero()) {\n
return digits + result;\n } else {\n
while (digits.length < 6) {\n
digits = '0' + digits;\n
}\n
result = '' + digits + result;\n }\n }\n};\n\n\n/**
@return {number} The high 32-bits as a signed value. */\nKotlin.Long.prototype.getHighBits = function() {\n
return this.high_;\n};\n\n\n/** @return {number} The low 32-bits as a signed value.
*/\nKotlin.Long.prototype.getLowBits = function() {\n return this.low_;\n};\n\n\n/** @return {number} The low
32-bits as an unsigned value. */\nKotlin.Long.prototype.getLowBitsUnsigned = function() {\n return (this.low_ >=
0) ?\n
this.low_ : Kotlin.Long.TWO_PWR_32_DBL_ + this.low_;\n};\n\n\n/**\n * @return {number} Returns
the number of bits needed to represent the absolute\n * value of this Long.\n
*/\nKotlin.Long.prototype.getNumBitsAbs = function() {\n if (this.isNegative()) {\n if
(this.equalsLong(Kotlin.Long.MIN_VALUE)) {\n
return 64;\n } else {\n
return
this.negate().getNumBitsAbs();\n }\n } else {\n var val = this.high_ != 0 ? this.high_ : this.low_;\n for (var bit
= 31; bit > 0; bit--) {\n
if ((val & (1 << bit)) != 0) {\n
break;\n
}\n }\n return this.high_ != 0 ? bit + 33
: bit + 1;\n }\n};\n\n\n/** @return {boolean} Whether this value is zero. */\nKotlin.Long.prototype.isZero =
function() {\n return this.high_ == 0 && this.low_ == 0;\n};\n\n\n/** @return {boolean} Whether this value is
negative. */\nKotlin.Long.prototype.isNegative = function() {\n return this.high_ < 0;\n};\n\n\n/** @return
{boolean} Whether this value is odd. */\nKotlin.Long.prototype.isOdd = function() {\n return (this.low_ & 1) ==
1;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return {boolean} Whether this
Long equals the other.\n */\nKotlin.Long.prototype.equalsLong = function(other) {\n return (this.high_ ==
other.high_) && (this.low_ == other.low_);\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare
against.\n * @return {boolean} Whether this Long does not equal the other.\n
*/\nKotlin.Long.prototype.notEqualsLong = function(other) {\n return (this.high_ != other.high_) || (this.low_ !=
other.low_);\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return {boolean}
Whether this Long is less than the other.\n */\nKotlin.Long.prototype.lessThan = function(other) {\n return
this.compare(other) < 0;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to compare against.\n * @return
{boolean} Whether this Long is less than or equal to the other.\n */\nKotlin.Long.prototype.lessThanOrEqual =
function(other) {\n return this.compare(other) <= 0;\n};\n\n\n/**\n * @param {Kotlin.Long} other Long to
compare against.\n * @return {boolean} Whether this Long is greater than the other.\n

Open Source Used In DNAC Platform Arica 9381


Open Source Used In DNAC Platform Arica
} else { // At this point, we have |other| >= 2, so |this/other| < |MIN_VALUE|
    var halfThis =
        this.shiftRight(1);
    var approx = halfThis.div(other).shiftLeft(1);
    if (approx.equalsLong(Kotlin.Long.ZERO)) { // return other.isNegative() ? Kotlin.Long.ONE :
        Kotlin.Long.NEG_ONE;
    } else { // var rem = this.subtract(other.multiply(approx));
        var result =
            approx.add(rem.divide(other));
        return result;
    } else if (other.equalsLong(Kotlin.Long.ZERO)) { // if (this.isNegative()) {
        return other.isNegative() ? Kotlin.Long.ONE :
            Kotlin.Long.NEG_ONE;
    } else { // var rem = this.subtract(other.multiply(approx));
        var result =
            approx.add(rem.divide(other));
        return result;
    }
    if (other.isNegative()) { // Kotlin.Long.ZERO;
        return Kotlin.Long.ZERO;
    }

    if (this.isNegative()) {
        if (other.isNegative()) {
            return this.negate().divide(other.negate());
        } else {
            return this.negate().divide(other).negate();
        }
    } else if (other.isNegative()) { // Kotlin.Long.ZERO;
        return this.divide(other.negate()).negate();
    }

    // Repeat the following until the remainder is less than other: find a
    // floating-point that approximates remainder / other *from below*, add this
    // into the result, and subtract it from the remainder.
    if (isCritical then // the approximate value is less than or equal to the real value so that the
    // remainder never becomes negative.
        var res = Kotlin.Long.ZERO;
        while (rem.greaterThanOrEqual(other)) {
            var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));
            var log2 = Math.ceil(Math.log(approx) / Math.LN2);
            var approxRem = approx.multiply(other);
            while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
                approx -= 1;
                approxRem = approx.multiply(other);
            }
            if (approxRem.isZero()) {
                approxRem = Kotlin.Long.ONE;
            }
            res = res.add(approxRem);
            rem = rem.subtract(approxRem);
        }
        return res;
    }

    /**
     * Returns this Long modulo the given one.
     * @param {Kotlin.Long} other Long by which to
     * mod.
     * @return {!Kotlin.Long} This Long modulo the given one.
     */
    Kotlin.Long.prototype.modulo =
    function(other) {
        return this.subtract(this.divide(other).multiply(other));
    }

    /** @return {!Kotlin.Long} The
     * bitwise-NOT of this value.
     */
    Kotlin.Long.prototype.not = function() {
        return Kotlin.Long.fromBits(~this.low_,
                                 ~this.high_);
    }

    /**
     * Returns the bitwise-AND of this Long and the given one.
     * @param {Kotlin.Long} other The Long with which to
     * AND.
     * @return {!Kotlin.Long} The bitwise-AND of this and the other.
     */
    Kotlin.Long.prototype.and = function(other) {
        return Kotlin.Long.fromBits(this.low_ & other.low_,
                                     this.high_ & other.high_);
    }

    /**
     * Returns the bitwise-XOR of this Long
     * and the given one.
     * @param {Kotlin.Long} other The Long with which to
     * XOR.
     * @return {!Kotlin.Long} The bitwise-XOR of this and the other.
     */
    Kotlin.Long.prototype.xor = function(other) {
        return Kotlin.Long.fromBits(this.low_ ^ other.low_,
                                     this.high_ ^ other.high_);
    }

    /**
     * Returns this Long with bits shifted to the left by the given amount.
     * @param {number} numBits The number of bits by which to shift.
     * @return {!Kotlin.Long} This shifted to the left by the given amount.
     */
    Kotlin.Long.prototype.shiftLeft = function(numBits) {
        return this.shiftLeft(numBits);
    }

    /**
     * Returns this Long with bits shifted to the right by the given amount.
     * @param {number} numBits The number of bits by which to shift.
     * @return {!Kotlin.Long} This shifted to the right by the given amount.
     */
    Kotlin.Long.prototype.shiftRight = function(numBits) {
        return this.shiftRight(numBits);
    }

    Open Source Used In DNAC Platform Arica 9383
/**
 * Returns this Long with bits shifted to the right by the given amount, with n * zeros placed into the new leading bits.
 * @param {number} numBits The number of bits by which to shift.
 * @return {Kotlin.Long} This shifted to the right by the given amount, with n * zeros placed into the new leading bits.
 */
Kotlin.Long.prototype.shiftRightUnsigned = function(numBits) {
  numBits &= 63;
  if (numBits === 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(
        (low >>> numBits) | (high << (32 - numBits)),
        high >>> numBits);
    } else if (numBits === 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >>> (numBits - 32), 0);
    }
  }
};

// Support for Kotlin
Kotlin.Long.prototype.equals = function (other) {
    return other instanceof Kotlin.Long &&
    this.equalsLong(other);
};

Kotlin.Long.prototype.compareTo = Kotlin.Long.prototype.compare;

Kotlin.Long.prototype.inc = function() {
    return this.add(Kotlin.Long.ONE);
};

Kotlin.Long.prototype.dec = function() {
    return this.add(Kotlin.Long.NEG_ONE);
};

Kotlin.Long.prototype.valueOf = function() {
    return this.toNumber();
};

Kotlin.Long.prototype.unaryPlus = function() {
    return this;
};

Kotlin.Long.prototype.unaryMinus = Kotlin.Long.prototype.negate;
Kotlin.Long.prototype.inv = Kotlin.Long.prototype.not;

Kotlin.Long.prototype.rangeTo = function (other) {
    return new Kotlin.kotlin.ranges.LongRange(this, other);
};

Kotlin.compareTo = function (a, b) {
    var typeA = typeof a;
    if (typeA === "number") {
        if (typeof b === "number") {
            return Kotlin.doubleCompareTo(a, b);
        }
        return Kotlin.primitiveCompareTo(a, b);
    }
    if (typeA === "string" || typeA === "boolean") {
        return Kotlin.primitiveCompareTo(a, b);
    }
    return a.compareTo_11rb$(b);
};

Kotlin.primitiveCompareTo = function (a, b) {
    return a < b ? -1 : a > b ? 1 : 0;
};

Kotlin.doubleCompareTo = function (a, b) {
    if (a < b) return -1;
    if (a > b) return 1;
    var ia = 1 / a;
    return ia === 1 / b ? 0 : (ia < 0 ? -1 : 1);
};

Kotlin.charInc = function (value) {
    return Kotlin.toChar(value+1);
};

Kotlin.charDec = function (value) {
    return Kotlin.toChar(value-1);
};

Kotlin.imul = Math.imul || imul;

Kotlin.imulEmulated = imul;

function imul(a, b) {
    return ((a & 0xffff0000) * (b & 0xffff) + (a & 0xffff) * (b | 0)) | 0;
}
bufFloat64 = new Float64Array(buf);
var bufFloat32 = new Float32Array(buf);
var bufInt32 = new Int32Array(buf);
var lowIndex = 0;
var highIndex = 1;

bufFloat64[0] = -1; // bff00000_00000000
if (bufInt32[lowIndex] !== 0) {
    lowIndex = 1;
    highIndex = 0;
}

Kotlin.getDoubleFromBits = function(value) {
    var bufFloat64[0] = value;
    return Kotlin.doubleToRawBits = function(value) {
        var bufFloat64[0] = value;
        return Kotlin.getDoubleFromBits = function(value) {
            var bufInt32[lowIndex] = value.low;
            var bufInt32[highIndex] = value.high;
            return bufFloat64[0];
        }
    }
}

Kotlin.doubleToBits = function(value) {
    var bufInt32[lowIndex] = value.low;
    var bufInt32[highIndex] = value.high;
    return bufFloat64[0];
}

Kotlin.floatToBits = function(value) {
    var bufInt32[lowIndex] = value;
    return bufFloat64[0];
}

Kotlin.floatToRawBits = function(value) {
    var bufInt32[lowIndex] = value;
    return bufFloat64[0];
}

Kotlin.doubleToRawBits = function(value) {
    var bufFloat64[0] = value;
    return Kotlin.getDoubleFromBits = function(value) {
        var bufInt32[lowIndex] = value;
        var Kotlin.ensureNotNaN = function(x) {
            return x != null;
        }
    }
}

if (typeof Math.asinh === "undefined") {
    var asinh = function(x) {
        if (x >= +taylor_n_bound)
            {
                if (x > upper_taylor_n_bound)
                    {
                        // approximation by laurent series in function(x) {
                return Math.log(x) + Math.LN2;
            }
        }
    }
}

if (typeof Math.tanh === "undefined") {
    Math.tanh = function(x) {
        if (Math.abs(x) < taylor_n_bound) {
            var result = x;
            if (Math.abs(x) > taylor_2_bound) {
                result -= (x * x * x) / 3;
                }
                return result;
            }
            else {
                var y = Math.exp(x);
                var y1 = 1 / y;
                if (isFinite(y)) return Math.exp(x - Math.LN2);
                if ((isFinite(y)) return Math.exp(x - Math.LN2);
                return (y - y1) / 2;
            }
        }
    }
}

if (typeof Math.cosh === "undefined") {
    Math.cosh = function(x) {
        var y = Math.exp(x);
        var y1 = 1 / y;
        if (isFinite(y) || !isFinite(y)) return Math.exp(Math.abs(x) - Math.LN2);
        return Math.sqrt(taylor_2_bound);
    }
}

var upper_taylor_2_bound = 1/taylor_2_bound;
var upper_taylor_n_bound = 1/taylor_n_bound;
if (Math.abs(x) < taylor_n_bound) {
    var result = x;
    if (Math.abs(x) > taylor_2_bound) {
        result += (x * x * x) / 6;
    }
    return result;
} else {
    var y = Math.exp(x);
    var y1 = 1 / y;
    if (isFinite(y)) return Math.exp(x - Math.LN2);
    if ((isFinite(y)) return Math.exp(x - Math.LN2);
    return (y - y1) / 2;
}

if (Math.abs(x) < taylor_n_bound) {
    var result = x;
    if (Math.abs(x) > taylor_2_bound) {
        result -= (x * x * x) / 3;
    }
    return result;
} else {
    var a = Math.exp(x), b = Math.exp(-x);
    return a === Infinity ? 1 : Math.log(a / b);
}

if (y > upper_taylor_2_bound) {
    return approximation by laurent series in 1/x at 0+ order from -1 to 0
} else {
    return approximation by laurent series in 1/x at 0+ order from -1 to 1
}
constructor.$metadata$;

    var klassMetadata = klass.$metadata$;

    // In WebKit (JavaScriptCore) for some interfaces from
    // DOM typeof returns "object", nevertheless they can be used in RHS of instanceof
    if (klassMetadata.kind === Kotlin.Kind.OBJECT) {
        return object === klass;
    }

    var klassMetadata = klass.$metadata$;

    // In WebKit (JavaScriptCore) for some interfaces from
    // DOM typeof returns "object", nevertheless they can be used in RHS of instanceof
    if (klassMetadata.kind === Kotlin.Kind.OBJECT & &
    object.constructor !== null) {
        return isInheritanceFromInterface(object.constructor, klass);
    }

    return false;
}

Kotlin.isNumber = function (a) {
    return typeof a == "number" || a instanceof Kotlin.Long;
}

Kotlin.isChar = function (value) {
    return value instanceof Kotlin.BoxedChar
};

Kotlin.isComparable = function (value) {
    var type = typeof value;

    return type === "string" ||
           type === "boolean" ||
           Kotlin.isNumber(value) ||
           Kotlin.isType(value, Kotlin.kotlin.Comparable);
};

Kotlin.isCharSequence = function (value) {
    return typeof value === "string" ||
           Kotlin.isType(value, Kotlin.kotlin.CharSequence);
};

package kotlin

public class Enum<T : Enum<T>> : Comparable<Enum<T>> {
    @JsName("name$") private var _name: String = ""

    @JsName("ordinal$") private var _ordinal: Int = 0

    val name: String
        get() = _name

    val ordinal:
        get() = _ordinal

    override fun compareTo(other: Enum<T>) = ordinal.compareTo(other.ordinal)

    override fun equals(other: Any?) = this === other

    override fun hashCode(): Int =
        js("Kotlin.identityHashCode")(this)

    override fun toString() = name

    companion object
};
import kotlin.random.
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component1(): T {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component1(): ByteArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component1(): ShortArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun IntArray.component1(): IntArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun LongArray.component1(): LongArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component1(): FloatArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component1(): DoubleArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component1(): BooleanArray {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharArray.component1(): CharArray {
    return get(0)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component2(): T {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component2(): ByteArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component2(): ShortArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun IntArray.component2(): IntArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun LongArray.component2(): LongArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component2(): FloatArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component2(): DoubleArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component2(): BooleanArray {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharArray.component2(): CharArray {
    return get(1)
}
Open Source Used In DNAC Platform Arica  9390

[IndexOfOutOfBoundsException] except in Kotlin/JS

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component2(): Double {
    return get(1) *
}

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component2(): Boolean {
    return get(1) *
}

@kotlin.internal.InlineOnly
public inline operator fun CharArray.component2(): Char {
    return get(1) *
}

@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component3(): T {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component3(): Byte {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component3(): Short {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun IntArray.component3(): Int {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun LongArray.component3(): Long {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component3(): Float {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component3(): Double {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component3(): Boolean {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun CharArray.component3(): Char {
    return get(2) *
}

@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component4(): T {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component4(): Byte {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component4(): Short {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun IntArray.component4(): Int {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun LongArray.component4(): Long {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component4(): Float {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component4(): Double {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component4(): Boolean {
    return get(3) *
}

@kotlin.internal.InlineOnly
public inline operator fun CharArray.component4(): Char {
    return get(3) *
}

[IndexOfOutOfBoundsException] except in Kotlin/JS

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.

*Where the behavior is unspecified.
@kotlin.internal.InlineOnly
public inline operator fun LongArray.component4(): Long {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component4(): Float {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component4(): Double {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component4(): Boolean {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharArray.component4(): Char {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component5(): T {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component5(): Byte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component5(): Short {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun IntArray.component5(): Int {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun LongArray.component5(): Long {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component5(): Float {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component5(): Double {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component5(): Boolean {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharArray.component5(): Char {
    return get(4)
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun ByteArray.contains(element: Byte): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun FloatArray.contains(element: Float): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun DoubleArray.contains(element: Double): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun BooleanArray.contains(element: Boolean): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if 
 * [element] is found in the array.
 */
public operator fun CharArray.contains(element: Char): Boolean {
    return indexOf(element) >= 0
}
[]
samples.collections.Collections.Elements.elementAtOrNull

BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNonNull(index)
} *\n
Returns an element at the given [index] or ‘null’ if the [index] is out of bounds of this array.

CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNonNull(index)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNonNull(index)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

CharArray.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

ByteBuffer.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

ByteBuffer.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

ShortBuffer.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

ShortBuffer.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

IntBuffer.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

IntBuffer.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

LongBuffer.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
} *\n
Returns the first element matching the given [predicate], or ‘null’ if no such element was found.

LongBuffer.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
} *\n
Returns the last element matching the given [predicate], or ‘null’ if no such element was found.

Open Source Used In DNAC Platform Arica 9394
FloatArray.findLast(predicate: Float -> Boolean): Float? {
    return lastOrNull(predicate)
}

@kotlin.internal.InlineOnly
public inline fun FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws [NoSuchElementException] if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws [NoSuchElementException] if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @throws [NoSuchElementException] if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun <T> Array<out T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ByteArray.first(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ShortArray.first(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun IntArray.first(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun LongArray.first(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun FloatArray.first(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun DoubleArray.first(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun BooleanArray.first(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun CharArray.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
element matching the given \[predicate\].\n* @throws \[NoSuchElementException\] if no such element is found.\n
    *\n    public inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {\n        for (element in this) if (predicate(element)) return element
    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    public inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {\n        for (element in this) if (predicate(element)) return element
    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {\n        for (element in this) if (predicate(element)) return element
    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public inline fun CharArray.first(predicate: (Char) -> Boolean): Char {\n        for (element in this) if (predicate(element)) return element
    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public <T> Array<out T>.firstOrNull(): T? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public ByteArray.firstOrNull(): Byte? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public ShortArray.firstOrNull(): Short? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public IntArray.firstOrNull(): Int? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public LongArray.firstOrNull(): Long? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public FloatArray.firstOrNull(): Float? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public DoubleArray.firstOrNull(): Double? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public BooleanArray.firstOrNull(): Boolean? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public CharArray.firstOrNull(): Char? {\n        if (isEmpty()) return null else this[0]\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public <T> Array<out T>.firstOrNull(predicate: (T) -> Boolean): T? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {\n        for (element in this) if (predicate(element)) return element\n    }\n    \n    throws \[NoSuchElementException\] if no such element is found.\n    *\n    public DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {\n        for (element in this) if (predicate(element)) return element\n    }
Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

```kotlin
Boolean): Char? {
    for (element in this) if (predicate(element)) return element
    return null
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun <T> Array<T>.orElseGet(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
public fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun CharArray.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ByteArray.getOrNull(index: Int): Byte? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ShortArray.getOrNull(index: Int): Short? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun IntArray.getOrNull(index: Int): Int? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun LongArray.getOrNull(index: Int): Long? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun FloatArray.getOrNull(index: Int): Float? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun DoubleArray.getOrNull(index: Int): Double? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun CharArray.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ByteArray.getOrNull(index: Int): Byte? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun ShortArray.getOrNull(index: Int): Short? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun IntArray.getOrNull(index: Int): Int? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun LongArray.getOrNull(index: Int): Long? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun FloatArray.getOrNull(index: Int): Float? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun DoubleArray.getOrNull(index: Int): Double? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```

```kotlin
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
public fun CharArray.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}
```
IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

ByteArray.lastIndexOf(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun isEmpty(): Boolean {
    if (predicate(this[index])) {
        return index
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> Array<out T>.indexOfLast(predicate: (T) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun <T> Array<out T>.indexOfLast(predicate: (T) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun ByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int { 
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}
this[lastIndex]\n} \n\n /**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ShortArray.last(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun IntArray.last(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun LongArray.last(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun FloatArray.last(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun DoubleArray.last(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun BooleanArray.last(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun CharArray.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun <T> Array<out T>.last(predicate: (T) -> Boolean): T {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ByteArray.last(predicate: (Byte) -> Boolean): Byte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ShortArray.last(predicate: (Short) -> Boolean): Short {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun IntArray.last(predicate: (Int) -> Boolean): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun LongArray.last(predicate: (Long) -> Boolean): Long {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun FloatArray.last(predicate: (Float) -> Boolean): Float {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun DoubleArray.last(predicate: (Double) -> Boolean): Double {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun BooleanArray.last(predicate: (Boolean) -> Boolean): Boolean {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun CharArray.last(predicate: (Char) -> Boolean): Char {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
### Open Source Used In DNAC Platform Arica 9401

```java
/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 * @since [1.4]
 */
public inline fun CharArray.last(predicate: (Char) -> Boolean): Char {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {
    if (isEmpty()) return -1
    var index = this.size - 1
    while (index >= 0) {
        if (element == this[index]) return index
        index--
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun ByteArray.lastIndexOf(element: Byte): Int {
    if (isEmpty()) return -1
    var index = this.size - 1
    while (index >= 0) {
        if (element == this[index]) return index
        index--
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun DoubleArray.lastIndexOf(element: Double): Int {
    for (index in this.indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun FloatArray.lastIndexOf(element: Float): Int {
    for (index in this.indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun IntArray.lastIndexOf(element: Int): Int {
    for (index in this(indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun LongArray.lastIndexOf(element: Long): Int {
    for (index in this(indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun ShortArray.lastIndexOf(element: Short): Int {
    for (index in this(indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun ByteArray.lastIndexOf(element: Byte): Int {
    for (index in this(indices.reversed()) {
        if (element == this[index]) return index
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain [element].
 */
public fun <T> Array<out T>.lastIndexOf(element: T): Int {
    if (isEmpty()) return -1
    var index = this.size - 1
    while (index >= 0) {
        if (element == this[index]) return index
        index--
    }
    return -1
}
```

---

**Open Source Used In DNAC Platform Arica 9401**

```
public function isEmpty(): boolean {
    return (this.size == 0)
}

public function size(): number {
    return this.length
}
```

---

**Open Source Used In DNAC Platform Arica 9401**

```
public function last(): any {
    return if (isEmpty()) null else this[this.size - 1]
}
```

---

**Open Source Used In DNAC Platform Arica 9401**

```
public function lastOrNull(): any? {
    return if (isEmpty()) null else this[this.size - 1]
}
```

---

**Open Source Used In DNAC Platform Arica 9401**

```
/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 * @since [1.4]
 */
public inline fun CharArray.last(predicate: (Char) -> Boolean): Char {
    for (index in this(indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    } throw NoSuchElementException("Array contains no element matching the predicate.")
}
```
public fun CharArray.lastOrNull(): Char? {
    return if (isEmpty()) null else this[0len - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> Array<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Array<T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ByteArray.random(): Byte {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ShortArray.random(): Short {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntArray.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongArray.random(): Long {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun FloatArray.random(): Float {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.random(): Double {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.random(): Boolean {
    return random(Random)
}

Open Source Used In DNA Platform Arica 9402
from this array.

@throws NoSuchElementException if this array is empty.

private fun CharArray.random(): Char
    return random(Random())

private fun <T> Array<T>.random(random: Random): T
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun <T> Array<out T>.random(random: Random): T
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun ByteArray.random(random: Random): Byte
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun ShortArray.random(random: Random): Short
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun IntArray.random(random: Random): Int
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun LongArray.random(random: Random): Long
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun FloatArray.random(random: Random): Float
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun DoubleArray.random(random: Random): Double
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun BooleanArray.random(random: Random): Boolean
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun CharArray.random(random: Random): Char
    if (isEmpty()) throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))

public fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

public fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

public fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

public fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

public fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

public fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

public fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

public fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())

private fun CharArray.randomOrNull(): Char?
    return randomOrNull(Random())

private fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

private fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

private fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

private fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

private fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

private fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

private fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

private fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())

public fun CharArray.randomOrNull(): Char?
    return randomOrNull(Random())

public fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

public fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

public fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

public fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

public fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

public fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

public fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

public fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())

private fun CharArray.randomOrNull(): Char?
    return randomOrNull(Random())

private fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

private fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

private fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

private fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

private fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

private fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

private fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

private fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())

public fun CharArray.randomOrNull(): Char?
    return randomOrNull(Random())

public fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

public fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

public fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

public fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

public fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

public fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

public fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

public fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())

private fun CharArray.randomOrNull(): Char?
    return randomOrNull(Random())

private fun <T> Array<out T>.randomOrNull(): T?
    return randomOrNull(Random())

private fun ByteArray.randomOrNull(): Byte?
    return randomOrNull(Random())

private fun ShortArray.randomOrNull(): Short?
    return randomOrNull(Random())

private fun IntArray.randomOrNull(): Int?
    return randomOrNull(Random())

private fun LongArray.randomOrNull(): Long?
    return randomOrNull(Random())

private fun FloatArray.randomOrNull(): Float?
    return randomOrNull(Random())

private fun DoubleArray.randomOrNull(): Double?
    return randomOrNull(Random())

private fun BooleanArray.randomOrNull(): Boolean?
    return randomOrNull(Random())


inline fun LongArray.randomOrNull(): Long? {
    return randomOrNull(Random) // Returns a random element from this array, or `null` if this array is empty.
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun FloatArray.randomOrNull(): Float? {
    return randomOrNull(Random) // Returns a random element from this array, or `null` if this array is empty.
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun DoubleArray.randomOrNull(): Double? {
    return randomOrNull(Random) // Returns a random element from this array, or `null` if this array is empty.
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun BooleanArray.randomOrNull(): Boolean? {
    return randomOrNull(Random) // Returns a random element from this array, or `null` if this array is empty.
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharArray.randomOrNull(): Char? {
    return randomOrNull(Random) // Returns a random element from this array, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T> Array<out T>.randomOrNull(random: Random): T? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun ByteArray.randomOrNull(random: Random): Byte? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun ShortArray.randomOrNull(random: Random): Short? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun IntArray.randomOrNull(random: Random): Int? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun LongArray.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun FloatArray.randomOrNull(random: Random): Float? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun DoubleArray.randomOrNull(random: Random): Double? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun BooleanArray.randomOrNull(random: Random): Boolean? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}

/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun CharArray.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return get(random.nextInt(size)) // Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
}
get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun <T> Array<out T>.single(): T {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun ByteArray.single(): Byte {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun ShortArray.single(): Short {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun IntArray.single(): Int {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun LongArray.single(): Long {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun FloatArray.single(): Float {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun DoubleArray.single(): Double {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun BooleanArray.single(): Boolean {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun CharArray.single(): Char {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw
            IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {
    var single: Byte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {
    var single: Short? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun CharArray.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given [predicate], or throws exception if there is no or more than one matching element.")
}
single element matching the given [predicate], or throws exception if there is no or more than one matching element.

```kotlin
*\public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Int
}

*\public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Long
}

*\public inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Float
}

*\public inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Double
}

*\public inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Boolean
}

*\public inline fun CharArray.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Char
}

*\public fun <T> Array<out T>.singleOrNull(): T? {
    return if (size == 1) this[0] else null
}
```

---

Open Source Used In DNAC Platform Arica 9406
element, or `null` if the array is empty or has more than one element.

```kotlin
open fun BooleanArray.singleOrNull():
    Boolean { 
        return if (size == 1) this[0] else null
    }

open fun CharArray.singleOrNull():
    Char? { 
        return if (size == 1) this[0] else null
    }

open funByteArray.singleOrNull():
    ByteArray? { 
        return if (size == 1) this[0] else null
    }

open fun ShortArray.singleOrNull(predicate: (Short) -> Boolean):
    Short? { 
        var single: Short? = null
        var found = false
        for (element in this) { 
            if (predicate(element)) { 
                if (found) return null
                single = element
                found = true
            } 
        } 
        if (!found) return null
        return single
    }

open fun IntArray.singleOrNull(predicate: (Int) -> Boolean):
    Int? { 
        var single: Int? = null
        var found = false
        for (element in this) { 
            if (predicate(element)) { 
                if (found) return null
                single = element
                found = true
            } 
        } 
        if (!found) return null
        return single
    }

open fun LongArray.singleOrNull(predicate: (Long) -> Boolean):
    Long? { 
        var single: Long? = null
        var found = false
        for (element in this) { 
            if (predicate(element)) { 
                if (found) return null
                single = element
                found = true
            } 
        } 
        if (!found) return null
        return single
    }

open fun ByteArray.singleOrNull(predicate: (Byte) -> Boolean):
    Byte? { 
        var single: Byte? = null
        var found = false
        for (element in this) { 
            if (predicate(element)) { 
                if (found) return null
                single = element
                found = true
            } 
        } 
        if (!found) return null
        return single
    }

open fun BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean):
    Boolean? { 
        return if (size == 1) this[0] else null
    }

open fun ByteArray.singleOrNull(predicate: (ByteArray) -> Boolean):
    ByteArray? { 
        return if (size == 1) this[0] else null
    }

open fun IntArray.singleOrNull(predicate: (IntArray) -> Boolean):
    IntArray? { 
        return if (size == 1) this[0] else null
    }
```

Open Source Used In DNAC Platform Arica 9407
require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n
/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.dropLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n
/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.dropLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n
/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.dropLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n
/**
 * Returns a list containing all elements except last [n] elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun <T>
    Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun ShortArray.dropLastWhile(predicate: (Short) -> Boolean): List<Short> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun IntArray.dropLastWhile(predicate: (Int) -> Boolean): List<Int> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun LongArray.dropLastWhile(predicate: (Long) -> Boolean): List<Long> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun FloatArray.dropLastWhile(predicate: (Float) -> Boolean): List<Float> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropLastWhile
 */
public inline fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}
Returns a list containing all elements except first elements that satisfy the given [predicate].

```kotlin
inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

The `dropWhile` function iterates through the array, adding elements to the list if the current element does not satisfy the given predicate. Once a non-satisfying element is encountered, the loop stops, and the list is returned. This behavior is useful for filtering out elements that meet certain criteria, allowing for efficient processing of large datasets.
a list containing only elements matching the given [predicate].

```kotlin
samples.collections.Collections.Filtering.filter

List<Byte> \{ \n  return filterTo(ArrayList<Byte>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Short> \{ \n  return filterTo(ArrayList<Short>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Int> \{ \n  return filterTo(ArrayList<Int>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Long> \{ \n  return filterTo(ArrayList<Long>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Float> \{ \n  return filterTo(ArrayList<Float>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Double> \{ \n  return filterTo(ArrayList<Double>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Boolean> \{ \n  return filterTo(ArrayList<Boolean>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Char> \{ \n  return filterTo(ArrayList<Char>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filter

List<Array<out T>> \{ \n  return filterIndexedTo(ArrayList<Array<out T>>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<ByteArray> \{ \n  return filterIndexedTo(ArrayList<ByteArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<ShortArray> \{ \n  return filterIndexedTo(ArrayList<ShortArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<IntArray> \{ \n  return filterIndexedTo(ArrayList<IntArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<LongArray> \{ \n  return filterIndexedTo(ArrayList<LongArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<FloatArray> \{ \n  return filterIndexedTo(ArrayList<FloatArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<DoubleArray> \{ \n  return filterIndexedTo(ArrayList<DoubleArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<BooleanArray> \{ \n  return filterIndexedTo(ArrayList<BooleanArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<CharArray> \{ \n  return filterIndexedTo(ArrayList<CharArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<Array<out T>> \{ \n  return filterIndexedTo(ArrayList<Array<out T>>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<ByteArray> \{ \n  return filterIndexedTo(ArrayList<ByteArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<ShortArray> \{ \n  return filterIndexedTo(ArrayList<ShortArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<IntArray> \{ \n  return filterIndexedTo(ArrayList<IntArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<LongArray> \{ \n  return filterIndexedTo(ArrayList<LongArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<FloatArray> \{ \n  return filterIndexedTo(ArrayList<FloatArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<DoubleArray> \{ \n  return filterIndexedTo(ArrayList<DoubleArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<BooleanArray> \{ \n  return filterIndexedTo(ArrayList<BooleanArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed

List<CharArray> \{ \n  return filterIndexedTo(ArrayList<CharArray>(), predicate) \n\} \n\n* @sample samples.collections.Collections.Filtering.filterIndexed
```

**Open Source Used In DNAC Platform Arica 9411**
given [predicate].

* @param [predicate] function that takes the index of an element and the element itself
* and returns the result of predicate evaluation on the element.

* @sample samples.collections.Collections.Filtering.filterIndexedTo

*//*public inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> {
    return filterIndexedTo(ArrayList<Double>(), predicate)
}

*//*

*//*public inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> {
    return filterIndexedTo(ArrayList<Boolean>(), predicate)
}

*//*

*//*public inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> {
    return filterIndexedTo(ArrayList<Char>(), predicate)
}

*//*

*//*public inline fun <T, C : MutableCollection<in T>> Array<out T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

*//*

*//*public inline fun <C : MutableCollection<in Byte>> ByteArray.filterIndexedTo(destination: C, predicate: (index: Int, Byte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

*//*

*//*public inline fun <C : MutableCollection<in Short>> ShortArray.filterIndexedTo(destination: C, predicate: (index: Int, Short) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

*//*

*//*public inline fun <C : MutableCollection<in Int>> IntArray.filterIndexedTo(destination: C, predicate: (index: Int, Int) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

*//*

*//*public inline fun <C : MutableCollection<in Long>> LongArray.filterIndexedTo(destination: C, predicate: (index: Int, Long) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

*//*

*//*public inline fun <C : MutableCollection<in Float>> FloatArray.filterIndexedTo(destination: C, predicate: (index: Int, Float) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}
samples.collections.Collections.Filtering.filterIndexedTo

/*\npublic inline fun <C : MutableCollection<in Double>> DoubleArray.filterIndexedTo(destination: C, predicate: (index: Int, Double) -> Boolean): C {\nforEachIndexed { index, element -> if (predicate(index, element)) destination.add(element)\n} return destination\n\n}@param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexedTo
*/

public inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterIndexedTo(destination: C, predicate: (index: Int, Boolean) -> Boolean): C {\nforEachIndexed { index, element -> if (predicate(index, element)) destination.add(element)\n} return destination\n\n}@param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexedTo
CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {\nforEachIndexed { index, element -> if (predicate(index, element)) destination.add(element)\n} return destination\n\n}@param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIndexedTo

/**
 * Returns a list containing all elements that are instances of specified type parameter R.\n* @param [predicate] function that takes the element as its parameter and returns the result of predicate evaluation on the element.\n* @sample samples.collections.Collections.Filtering.filterIsInstanceTo
*/

public inline fun <reified R> Array<*>.filterIsInstanceTo(destination: MutableList<R>): List<R> {
    filterIsInstanceTo(destination)\n\n}@sample samples.collections.Collections.Filtering.filterIsInstanceTo

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {
    filterNotTo(ArrayList<T>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {
    filterNotTo(ArrayList<Byte>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {
    filterNotTo(ArrayList<Short>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {
    filterNotTo(ArrayList<Int>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {
    filterNotTo(ArrayList<Long>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {
    filterNotTo(ArrayList<Float>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double> {
    filterNotTo(ArrayList<Double>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean): List<Boolean> {
    filterNotTo(ArrayList<Boolean>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements not matching the given [predicate].\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun CharArray.filterNot(predicate: (Char) -> Boolean): List<Char> {
    filterNotTo(ArrayList<Char>(), predicate)\n\n}@sample samples.collections.Collections.Filtering.filter

/**
 * Returns a list containing all elements that are not `null`.\n* @sample samples.collections.Collections.Filtering.filter
*/

public inline fun Any.filterNotNull(): Any {\n    this\n\n}@sample samples.collections.Collections.Filtering.filter

fun <T : Any> Array<out T?>.filterNotNull(): List<T> { 
    return filterNotNullTo(ArrayList<T>())
}

fun <T : Any> Array<out T?>.filterNotNullTo(destination: C): C { 
    for (element in this) if (element != null) 
        destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> 
    Array<out T?>.filterNotTo(destination: C, predicate: (T) -> Boolean): C { 
    for (element in this) if (!predicate(element)) 
        destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <T, C : MutableCollection<in T>> Array<out T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C { 
    for (element in this) if (!predicate(element)) 
        destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <T, C : MutableCollection<in T>> Array<out T>.forEachTo(destination: C, predicate: (T) -> Boolean): C { 
    for (element in this) if (predicate(element)) 
        destination.add(element)
    return destination
}
public inline fun <C : MutableCollection<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public inline fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun ShortArray.slice(indices: IntRange): List<Short> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun CharArray.slice(indices: IntRange): List<Char> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun <T> Array<out T>.slice(indices: Iterable<Int>): List<T> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<T>(size)
    for
BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {
    val result = BooleanArray(indices.size)
    return result
}

/**
 * Returns an array containing elements at specified [indices].
 */
public fun IntArray.slice(indices: Iterable<Int>): List<Int> {
    val result = ArrayList<Int>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun ShortArray.slice(indices: Iterable<Int>): List<Short> {
    val result = ArrayList<Short>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun LongArray.slice(indices: Iterable<Int>): List<Long> {
    val result = ArrayList<Long>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun FloatArray.slice(indices: Iterable<Int>): List<Float> {
    val result = ArrayList<Float>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun ByteArray.slice(indices: Iterable<Int>): List<Byte> {
    val result = ArrayList<Byte>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun CharArray.slice(indices: Iterable<Int>): List<Char> {
    val result = ArrayList<Char>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}
var targetIndex = 0
for (sourceIndex in indices) {
    result[targetIndex++] = this[sourceIndex]
}

return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun <T> Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun CharArray.sliceArray(indices: IntRange): CharArray {
    if (indices.isEmpty()) return CharArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements of this array at specified [indices] range.
 */
public fun CharArray.sliceArray(indices: Collection<Int>): CharArray {
    val result = CharArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun DoubleArray.sliceArray(indices: IntRange): DoubleArray {
    if (indices.isEmpty()) return DoubleArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun DoubleArray.sliceArray(indices: Collection<Double>): DoubleArray {
    var result = DoubleArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun FloatArray.sliceArray(indices: IntRange): FloatArray {
    if (indices.isEmpty()) return FloatArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun FloatArray.sliceArray(indices: Collection<Float>): FloatArray {
    val result = FloatArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun LongArray.sliceArray(indices: IntRange): LongArray {
    if (indices.isEmpty()) return LongArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun LongArray.sliceArray(indices: Collection<Long>): LongArray {
    var result = LongArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun IntArray.sliceArray(indices: IntRange): IntArray {
    if (indices.isEmpty()) return IntArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun IntArray.sliceArray(indices: Collection<Int>): IntArray {
    var result = IntArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun ShortArray.sliceArray(indices: IntRange): ShortArray {
    if (indices.isEmpty()) return ShortArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun ShortArray.sliceArray(indices: Collection<Short>): ShortArray {
    var result = ShortArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun ByteArray.sliceArray(indices: IntRange): ByteArray {
    if (indices.isEmpty()) return ByteArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun ByteArray.sliceArray(indices: Collection<Byte>): ByteArray {
    var result = ByteArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun <T> Array<out T>.sliceArray(indices: IntRange): Array<out T> {
    if (indices.isEmpty()) return copyOfRange(0, 0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
public fun <T> Array<T>.sliceArray(indices: Collection<T>): Array<T> {
    if (indices.isEmpty()) return copyOfRange(0, 0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}
samples.collections.Collections.Transformations.take

public fun LongArray.take(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Long>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

public fun IntArray.takeLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Int>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

public fun ShortArray.takeLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Short>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

public fun ByteArray.takeLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Byte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

public fun <T> Array<out T>.takeLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

public fun BooleanArray.take(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Boolean>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

public fun DoubleArray.take(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Double>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

public fun FloatArray.take(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Float>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

public fun CharArray.take(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Char>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}
Unit
val midPoint = (size / 2) - 1
if (midPoint < 0) return
var reverseIndex = lastIndex
for (index in 0..midPoint)
    val tmp = this[index]
    this[index] = this[reverseIndex]
    this[reverseIndex] = tmp
    reverseIndex--
}

/**
 * Reverses elements in the array in-place.
 */
public fun IntArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
}

/**
 * Reverses elements in the array in-place.
 */
public fun ByteArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun CharArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun BooleanArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun DoubleArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun FloatArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun LongArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint)
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}
toIndex - 1
for (index in fromIndex until midPoint) {
    val tmp = this[index]
    this[index] = this[reverseIndex]
    this[reverseIndex] = tmp
    reverseIndex--
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun LongArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun FloatArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun DoubleArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements of the array in the specified range in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to reverse.
 * @param toIndex the end of the range (exclusive) to reverse.
 * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.4")
public fun BooleanArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val midPoint = (fromIndex + toIndex) / 2
    if (fromIndex == midPoint) return
    var reverseIndex = toIndex - 1
    for (index in fromIndex until midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Array<out T>.reversed(): List<T> {
    if (isEmpty())
        return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun ByteArray.reversed(): List<Byte> {
    if (isEmpty())
        return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun ShortArray.reversed(): List<Short> {
    if (isEmpty())
        return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun CharArray.reversed(): List<Char> {
    if (isEmpty())
        return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
```kotlin
*//public fun IntArray.reversed(): List<Int> { // if (isEmpty()) return emptyList() // val list = toMutableList() // list.reverse() // return list // } // *//

/**
 * Returns a list with elements in reversed order.
 */
public fun IntArray.reversed(): List<Int> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun LongArray.reversed(): List<Long> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun FloatArray.reversed(): List<Float> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun DoubleArray.reversed(): List<Double> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun BooleanArray.reversed(): List<Boolean> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun CharArray.reversed(): List<Char> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun <T> Array<T>.reversedArray(): Array<T> {
    if (isEmpty()) return this
    val result = arrayOfNulls(this, size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun ByteArray.reversedArray(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun ShortArray.reversedArray(): ShortArray {
    if (isEmpty()) return this
    val result = ShortArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun IntArray.reversedArray(): IntArray {
    if (isEmpty()) return this
    val result = IntArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun LongArray.reversedArray(): LongArray {
    if (isEmpty()) return this
    val result = LongArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun FloatArray.reversedArray(): FloatArray {
    if (isEmpty()) return this
    val result = FloatArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun DoubleArray.reversedArray(): DoubleArray {
    if (isEmpty()) return this
    val result = DoubleArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun BooleanArray.reversedArray(): BooleanArray {
    if (isEmpty()) return this
    val result = BooleanArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun CharArray.reversedArray(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun <T> Array<T>.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun ByteArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun ShortArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun IntArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun LongArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun FloatArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun DoubleArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place.
 */
@SinceKotlin("1.4")
public fun BooleanArray.shuffle(): Unit {
    shuffle(Random)
}
```

Open Source Used In DNAC Platform Arica 9423
shuffles elements in this array in-place. */
@SinceKotlin("1.4")
public fun CharArray.shuffle(): Unit {
    Random.shuffle(this)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun <T> Array<T>.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun ByteArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun ShortArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun IntArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun LongArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun FloatArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun DoubleArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
public fun BooleanArray.shuffle(random: Random): Unit {
    Random.shuffle(this, random)
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 */
@SinceKotlin("1.4")
public fun <T> Array<T>.sort(selector: (T, T) -> Int): Unit {
    this.sortBy { selector }
other after sorting.

*/

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R): Unit {
    if (isEmpty()) return this
    sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public fun FloatArray.sortedArray(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun LongArray.sortedArray(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun IntArray.sortedArray(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun ShortArray.sortedArray(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun ByteArray.sortedArray(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun <T : Comparable<T>> Array<out T>.sortedArray(): List<T> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R): Unit {
    if (isEmpty()) return this
    sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public fun <T : Comparable<T>> Array<out T>.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun FloatArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun LongArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun IntArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun ShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun ByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place according to their natural sort order.
 */
public fun <T : Comparable<T>> Array<out T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareBy(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareBy(selector))
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

// Sorts elements in the array in-place sorting.

public inline fun <T, R : Comparable<R>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}
(isEmpty()) return this
    return this.copyOf().apply { sort().run()} // Create a sorted array.
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */

public fun DoubleArray.sortedArray(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() } // Sort the array.
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */

public fun CharArray.sortedArray(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() } // Sort the array.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public fun ByteArray.sortedArrayDescending(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */

public fun ShortArray.sortedArrayDescending(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */

public fun IntArray.sortedArrayDescending(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */

public fun LongArray.sortedArrayDescending(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */

public fun FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */

public fun CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Sort in descending order.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <T, R : Comparable<R>> Array<out T>.sortedBy(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareBy(selector)) // Sort the array.
}

/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */

public inline fun <R : Comparable<R>> CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareBy(selector)) // Sort the array.
}
LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

ShortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector))
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareBy(selector))
} // Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
List<T> {\n    return sortedWith(reverseOrder());\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun ByteArray.sortedDescending(): List<Byte> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun ShortArray.sortedDescending(): List<Short> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun IntArray.sortedDescending(): List<Int> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun LongArray.sortedDescending(): List<Long> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun FloatArray.sortedDescending(): List<Float> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun DoubleArray.sortedDescending(): List<Double> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted descending according to their natural sort order.

*public fun CharArray.sortedDescending(): List<Char> {\n    return copyOf().apply { sort() }.reversed();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*stable. It means that equal elements preserve their order relative to each other after sorting.

*public fun <T> Array<out T>.sortedWith(comparator: Comparator<in T>): List<T> {\n    return sortedArrayWith(comparator).asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun ByteArray.sortedWith(comparator: Comparator<in Byte>): List<Byte> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun ShortArray.sortedWith(comparator: Comparator<in Short>): List<Short> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun IntArray.sortedWith(comparator: Comparator<in Int>): List<Int> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun LongArray.sortedWith(comparator: Comparator<in Long>): List<Long> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun FloatArray.sortedWith(comparator: Comparator<in Float>): List<Float> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun BooleanArray.sortedWith(comparator: Comparator<in Boolean>): List<Boolean> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].

*public fun CharArray.sortedWith(comparator: Comparator<in Char>): List<Char> {\n    return toTypedArray().apply { sortWith(comparator) }.asList();\n} \n
* Returns a list of all elements sorted according to the specified [comparator].
For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

If two corresponding elements are nested arrays, they are also compared deeply.

* If any of arrays contains itself on any nesting level, the behavior is undefined.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* The specified arrays are also considered deeply equal if both are `null`. *

* If two corresponding elements are nested arrays, they are also compared deeply.

* Any parameters except one another.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* If any of arrays contains itself on any nesting level, the behavior is undefined.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* The specified arrays are also considered deeply equal if both are `null`. *

* If two

* The specified arrays are also considered deeply equal if both are `null`. *

* If two corresponding elements are nested arrays, they are also compared deeply.

* Any parameters except one another.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* The specified arrays are also considered deeply equal if both are `null`. *

* If two corresponding elements are nested arrays, they are also compared deeply.

* Any parameters except one another.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* The specified arrays are also considered deeply equal if both are `null`. *

* If two corresponding elements are nested arrays, they are also compared deeply.

* Any parameters except one another.

* The elements of other types are compared for equality with the `[equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. *

* The specified arrays are also considered deeply equal if both are `null`. *

* If two corresponding elements are nested arrays, they are also compared deeply.
floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun LongArray.contentEquals(other: LongArray): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.1")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public expect infix fun CharArray.contentEquals(other: CharArray): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun <T> Array<out T>? contentEquals(other: Array<out T>?): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun ShortArray?.contentEquals(other: ShortArray?): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun IntArray?.contentEquals(other: IntArray?): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun <T> ShortArray<out T>.contentEquals(other: ShortArray<out T>): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun <T> IntArray<out T>.contentEquals(other: IntArray<out T>): Boolean

/**
 * Returns 'true' if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order. The elements are compared for equality with the [equals][Any.equals] function. If for floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

@SinceKotlin("1.4")
public expect infix fun <T> LongArray<out T>.contentEquals(other: LongArray<out T>): Boolean
equal to '0.0'.

public expect infix fun FloatArray?.contentEquals(other: FloatArray?):
Boolean

* Returns `true` if the two specified arrays are *structurally* equal to one another.

* i.e. contain the same number of the same elements in the same order.

* The elements are compared for equality with the

* [equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.

public expect infix fun DoubleArray?.contentEquals(other: DoubleArray?):
Boolean

* Returns `true` if the two specified arrays are *structurally* equal to one another,

* i.e. contain the same number of the same elements in the same order.

* The elements are compared for equality with the

* [equals][Any.equals] function.

* For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.

public expect infix fun CharArray?.contentEquals(other: CharArray?):
Boolean

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun <T> Array<out T>.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun ByteArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun ShortArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun IntArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun LongArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun FloatArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun DoubleArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.

public expect fun CharArray.contentHashCode(): Int

* Returns a hash code based on the contents of this array as if it is

* [List].

* For Kotlin compiler 1.4 to avoid deprecation

* warning.
hash code based on the contents of this array as if it is [List].

@SinceKotlin("1.4")
public expect fun FloatArray?.contentHashCode(): Int?

@SinceKotlin("1.4")
public expect fun DoubleArray?.contentHashCode(): Int?

@SinceKotlin("1.4")
public expect fun BooleanArray?.contentHashCode(): Int?

@SinceKotlin("1.4")
public expect fun CharArray?.contentHashCode(): Int?

@SinceKotlin("1.4")
public expect fun <T> Array<T?>.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<out T>?.@sample

@SinceKotlin("1.4")
public expect fun ByteArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun ShortArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun IntArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun LongArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun FloatArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun DoubleArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun BooleanArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun CharArray.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<T>.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<out T>.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<out T>.contentToString(): String

@SinceKotlin("1.4")
public expect fun <T> Array<out T>.contentToString(): String
public expect fun ShortArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 * @since Kotlin("1.4")
 */
@SinceKotlin("1.4")
public expect fun IntArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
public expect fun LongArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
public expect fun FloatArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
public expect fun DoubleArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.4")
public expect fun BooleanArray?.contentToString(): String

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * @since Kotlin("1.3")
 */
@SinceKotlin("1.3")
public expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * @since Kotlin("1.3")
 */
@SinceKotlin("1.3")
public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray


It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndices the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndices the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @return the [destination] array.

* @since Kotlin("1.3")

public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndices: Int = 0, endIndices: Int = Int(size)): IntArray

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @return the [destination] array.

* @since Kotlin("1.3")

public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndices: Int = 0, endIndices: Int = size): LongArray

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @return the [destination] array.

* @since Kotlin("1.3")

public expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndices: Int = 0, endIndices: Int = size): FloatArray

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @return the [destination] array.

* @since Kotlin("1.3")

public expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndices: Int = 0, endIndices: Int = size): DoubleArray

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @return the [destination] array.

* @since Kotlin("1.3")

public expect fun
subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].

* or when that index is out of the [destination] array indices range.

* @return the [destination] array.

*/
@SinceKotlin("1.3")
public expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):

BooleanArray

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] array to copy to, 0 by default.

* @param destination the position in the [destination] array to copy to, 0 by default.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].

* or when that index is out of the [destination] array indices range.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @return the [destination] array.

CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):

CharArray

* Returns new array which is a copy of the original array.

* @return the [destination] array.

* @SinceKotlin("1.3")
public expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):

CharArray

* Returns new array which is a copy of the original array.

* @Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOf():

Array<T>

* Returns new array which is a copy of the original array.

* @param parameter the position in the [destination] array to copy to, 0 by default.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @return the [destination] array.

ByteArray.copyOf():

ByteArray

* Returns new array which is a copy of the original array.

* @sample samples.collections.Arrays.CopyOfOperations.copyOf

* @SinceKotlin("1.3")
public expect fun ByteArray.copyOf(newSize: Int):

ByteArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @param newSize the size of the new array.

* @return the new array.

ShortArray.copyOf(newSize: Int):

ShortArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @param newSize the size of the new array.

* @return the new array.

IntArray.copyOf():

IntArray

* Returns new array which is a copy of the original array.

* @sample samples.collections.Arrays.CopyOfOperations.copyOf

* @param newSize the size of the new array.

* @return the new array.

FloatArray.copyOf(newSize: Int):

FloatArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @param newSize the size of the new array.

* @return the new array.

DoubleArray.copyOf(newSize: Int):

DoubleArray

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

* @param newSize the size of the new array.

* @return the new array.
of the specified range of the original array.
*   * @param fromIndex the start of the range (inclusive) to copy.
*   * @param toIndex the end of the range (exclusive) to copy.
*   * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
*   * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray

LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray

FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray

DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray

BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray

CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray

ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.3")
public expect function
is greater than the size of this array.

* @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IllegalArgument Exception if [fromIndex] is greater than
[toIndex].

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IllegalArgumentException if [fromIndex] is greater than
[toIndex].

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.

* @throws IndexOutOfBoundsException if [fromIndex] is less than zero
or [toIndex] is greater than the size of this array.
Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun DoubleArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun BooleanArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun ByteArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun ShortArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun IntArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun LongArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun FloatArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun DoubleArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun BooleanArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun CharArray.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public inline fun <T> Array<out T>.isEmpty(): Boolean {\n    return size == 0\n} *\n
Returns `true` if the array is empty.\n
*\n
@kotlin.internal.InlineOnly\n
public expect operator fun <T> Array<out T>.plus(element: T): Array<T> *\n
Suppress("NO_ACTUAL_FOR_EXPECT")\n
Array<out T>.plus(element: T): Array<T> *\n
Array<out T>.plus(element: Byte): ByteArray *\n
Array<out T>.plus(element: Short): ShortArray *\n
Array<out T>.plus(element: Int): IntArray *\n
Array<out T>.plus(element: Long): LongArray *\n
Array<out T>.plus(elements: Collection<T>): Array<T> *\n
Array<out T>.plus(elements: Collection<Byte>): ByteArray *\n
Array<out T>.plus(elements: Collection<Short>): ShortArray *\n
Array<out T>.plus(elements: Collection<Int>): IntArray *\n
Array<out T>.plus(elements: Collection<Long>): LongArray *\n
Array<out T>.plus(elements: Collection<Float>): FloatArray *\n
Array<out T>.plus(elements: Collection<Double>): DoubleArray *\n
Array<out T>.plus(elements: Collection<Char>): CharArray *\n
Array<out T>.plus(elements: Collection<Byte>): ByteArray *\n
Array<out T>.plus(elements: Collection<Short>): ShortArray *\n
Array<out T>.plus(elements: Collection<Int>): IntArray *\n
Array<out T>.plus(elements: Collection<Long>): LongArray *\n
Array<out T>.plus(elements: Collection<Float>): FloatArray *\n
Array<out T>.plus(elements: Collection<Double>): DoubleArray *\n
Array<out T>.plus(elements: Collection<Char>): CharArray
original array and then all elements of the given [elements] collection.
 */
public expect operator fun ShortArray.plus(elements: Collection<Short>): ShortArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun IntArray.plus(elements: Collection<Int>): IntArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun LongArray.plus(elements: Collection<Long>): LongArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun FloatArray.plus(elements: Collection<Float>): FloatArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public expect operator fun CharArray.plus(elements: Collection<Char>): CharArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<T>.plus(elements: Array<out T>): Array<T>

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun ByteArray.plus(elements: ByteArray): ByteArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun ShortArray.plus(elements: ShortArray): ShortArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun IntArray.plus(elements: IntArray): IntArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun LongArray.plus(elements: LongArray): LongArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun FloatArray.plus(elements: FloatArray): FloatArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public expect operator fun CharArray.plus(elements: CharArray): CharArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.plusElement(element: T): Array<T>

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun IntArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun LongArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun ByteArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun ShortArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun DoubleArray.sort(): Unit

/**
 * Sorts the array in-place according to the natural order of its elements.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
@sample samples.collections.Arrays.Sorting.sortArrayOfComparable
@public expect fun <T : Comparable<T>> Array<T>.sort(): Unit

*/

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.plusElement(element: T): Array<T>

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun IntArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun LongArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun ByteArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun ShortArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public expect fun DoubleArray.sort(): Unit

/**
 * Sorts the array in-place according to the natural order of its elements.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
@sample samples.collections.Arrays.Sorting.sortArrayOfComparable
@public expect fun <T : Comparable<T>> Array<T>.sort(): Unit

*/
fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * @throws IndexOutOfBoundsException if [fromIndex] is greater than [toIndex].\n * @sample samples.collections.Arrays.Sorting.sortRangeOfArrayOfComparable\n */
@SinceKotlin("1.4")
public expect fun <T : Comparable<T>> Array<T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun ByteArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit

@SinceKotlin("1.4")
public expect fun <T : Comparable<T>> Array<T>.sortRangeOfArrayOfComparable\n
@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
Comparator<in T>): Unit

/**
 * Sorts a range in the array in-place with the given [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit

/**
 * Returns an array of Char containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toCharArray(): CharArray

/**
 * Returns an array of Double containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toDoubleArray(): DoubleArray

/**
 * Returns an array of Float containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toFloatArray(): FloatArray

/**
 * Returns an array of Int containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toIntArray(): IntArray

/**
 * Returns an array of Long containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toLongArray(): LongArray

/**
 * Returns an array of Short containing all of the elements of this generic array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun <T> Array<out T>.toShortArray(): ShortArray

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun CharArray.toTypedArray(): Array<Char>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun IntArray.toTypedArray(): Array<Int>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun LongArray.toTypedArray(): Array<Long>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun ShortArray.toTypedArray(): Array<Short>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
samples.collections.Arrays.Transformations.associateArrayOfPrimitives

ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

DoubleArray.associate(transform: (Double) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

Array<out T>.associateBy(keySelector: (T) -> K): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByKeyTo(LinkedHashMap<K, V>(capacity), keySelector)
}
Open Source Used In DNAC Platform Arica 9445

returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/public inline fun <K>  
ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Byte>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Short>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Int>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Long>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Float>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Double>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key
returned from [keySelector] function applied to each element. If any two elements would have the same key
returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
*/
BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean>  
val capacity = 
mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, Boolean>(capacity),
keySelector)

/* Returns a [Map] containing the elements from the given array indexed by the key

Open Source Used In DNAC Platform Arica 9445
return from [keySelector] function applied to each element. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/*public inline fun <K> CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, Char>(capacity),
  keySelector).un/*un */ Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

/*public inline fun <T, K, V> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, V>(capacity),
  keySelector, valueTransform).un/*un */ Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

/*public inline fun <K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, V>(capacity),
  keySelector, valueTransform).un/*un */ Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

/*public inline fun <T, K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, V>(capacity),
  keySelector, valueTransform).un/*un */ Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

/*public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, V>(capacity),
  keySelector, valueTransform).un/*un */ Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. @sample

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

/*public inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
  val capacity = mapCapacity(size).coerceAtLeast(16)
  return associateByTo(LinkedHashMap<K, V>(capacity),
keySelector, valueTransform) Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. 

If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array. 

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform) Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. 

If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform) Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. 

If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

fun <K, V> CharArray.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform) Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array. 

If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform

fun <T, K, M : MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element) 
    }
    return destination
}

fun <K, M : MutableMap<in K, in Byte>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element) 
    }
    return destination
}

fun <K, M : MutableMap<in K, in Short>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element) 
    }
    return destination
}

fun <K, M : MutableMap<in K, in Int>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element) 
    }
    return destination
}
two elements would have the same key returned by [keySelector] the last one gets added to the map.  

```kotlin
@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

public inline fun <K, M : MutableMap<in K, in Long>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

public inline fun <K, M : MutableMap<in K, in Float>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

public inline fun <K, M : MutableMap<in K, in Double>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

public inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
```

Open Source Used In DNAC Platform Arica  9448
K, valueTransform: (Long) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs,\n * where key is provided by the [keySelector] function and\n * and value is provided by the [valueTransform] function applied to elements of the given array.\n * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform
 */
public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}

/**
 * Populates
 * and returns the [destination] mutable map with key-value pairs
 * provided by [transform] function applied to each
 * element of the given array.
 * 
 * If any of two pairs would have the same key returned by [keySelector] the last one gets added to the map.
 * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateTo(destination: M, transform: (T) ->
Pair<K, V>): M {\n    for (element in this) {\n        destination += transform(element)\n    }\n    return destination\n}\n/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */ @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

/**\n * Populates and returns the [destination] mutable map with key-value pairs\n * provided by [transform] function applied to each element of the given array.\n * If any of two pairs would have the same key the last one gets added to the\n * map.\n */ @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

/**\n * Returns a [Map] where keys are elements from the given array and values are\n * produced by the [valueSelector] function applied to each element.\n */ @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo
@sample samples.collections.Collections.Transformations.associateWith

@SinceKotlin("1.4")
@sample samples.collections.Collections.Transformations.associateWith
fun <K, V> Array<out K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

@Sample {samples.collections.Collections.Transformations.associateWith
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <K, V> Array<out K>.associateWithTo(result: Map<K, V>, valueSelector: (K) -> V): Map<K, V> {
    var entry = result
    var lastEntry = result
    for (element in this)
        if (entry.containsKey(element))
            entry = entry.add(element, valueSelector(element))
        else
            lastEntry = entry.add(element, valueSelector(element))
    return lastEntry
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> ByteArray.associateWith(valueSelector: (Byte) -> V): Map<Byte, V> {
    val result = LinkedHashMap<Byte, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> ShortArray.associateWith(valueSelector: (Short) -> V): Map<Short, V> {
    val result = LinkedHashMap<Short, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> IntArray.associateWith(valueSelector: (Int) -> V): Map<Int, V> {
    val result = LinkedHashMap<Int, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> LongArray.associateWith(valueSelector: (Long) -> V): Map<Long, V> {
    val result = LinkedHashMap<Long, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> FloatArray.associateWith(valueSelector: (Float) -> V): Map<Float, V> {
    val result = LinkedHashMap<Float, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/*
 * Returns a [Map] where keys are elements from the given array and values are produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
public inline fun <V> DoubleArray.associateWith(valueSelector: (Double) -> V): Map<Double, V> {
    val result = LinkedHashMap<Double, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

Open Source Used In DNAC Platform Arica 9451
@sample samples.collections.Collections.Transformations.associateWithin

```kotlin
BooleanArray.associateWith(valueSelector: (Boolean) -> V): Map<Boolean, V> {
    val result =
    LinkedHashMap<Boolean, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}
```

```kotlin
/*
 * Returns a [Map] where keys are elements from the given array and values are:
 * produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V>

```kotlin
CharArray.associateWith(valueSelector: (Char) -> V): Map<Char, V> {
    val result =
    LinkedHashMap<Char, V>(mapCapacity(size.coerceAtMost(128)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

```kotlin
/*
 * Populates and returns the [destination] mutable map with key-value pairs for each
 * element of the given array.
 * Where key is the element itself and value is provided by the [valueSelector] function
 * applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Byte, in V>>

```kotlin
ByteArray.associateWithTo(destination: M, valueSelector: (Byte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

```kotlin
/*
 * Populates and returns the [destination] mutable map with key-value pairs for each
 * element of the given array.
 * Where key is the element itself and value is provided by the [valueSelector] function
 * applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Short, in V>>

```kotlin
ShortArray.associateWithTo(destination: M, valueSelector: (Short) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

```kotlin
/*
 * Populates and returns the [destination] mutable map with key-value pairs for each
 * element of the given array.
 * Where key is the element itself and value is provided by the [valueSelector] function
 * applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Int, in V>>

```kotlin
IntArray.associateWithTo(destination: M, valueSelector: (Int) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

```kotlin
/*
 * Populates and returns the [destination] mutable map with key-value pairs for each
 * element of the given array.
 * Where key is the element itself and value is provided by the [valueSelector] function
 * applied to that key.
 * If any two elements are equal, the last one overwrites the former value in the map.
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in Long, in V>>

```kotlin
LongArray.associateWithTo(destination: M, valueSelector: (Long) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}
```
samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M :MutableMap<in Float, in V>>
FloatArray.associateWithTo(destination: M, valueSelector: (Float) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the destination mutable map with key-value pairs for each element of the given array.
 * 
 * If any two elements are equal, the last one overwrites the former value in the map.
 * 
 * @sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M :MutableMap<in Double, in V>>
DoubleArray.associateWithTo(destination: M, valueSelector: (Double) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the destination mutable map with key-value pairs for each element of the given array.
 * 
 * If any two elements are equal, the last one overwrites the former value in the map.
 * 
 * @sample samples.collections.Collections.Transformations.associateWithTo

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <V, M :MutableMap<in Boolean, in V>>
BooleanArray.associateWithTo(destination: M, valueSelector: (Boolean) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <T, C :MutableCollection<in T>> Array<out T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun <T> Array<out T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}
fun ByteArray.toHashSet(): HashSet<Byte> {
    return toCollection(HashSet<Byte>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun ShortArray.toHashSet(): HashSet<Short> {
    return toCollection(HashSet<Short>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun IntArray.toHashSet(): HashSet<Int> {
    return toCollection(HashSet<Int>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun LongArray.toHashSet(): HashSet<Long> {
    return toCollection(HashSet<Long>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun FloatArray.toHashSet(): HashSet<Float> {
    return toCollection(HashSet<Float>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun DoubleArray.toHashSet(): HashSet<Double> {
    return toCollection(HashSet<Double>(mapCapacity(size)))
}

/**
 * Returns a new [HashSet] of all elements.
 */
public fun BooleanArray.toHashSet(): HashSet:Boolean> {
    return toCollection(HashSet<Char>(mapCapacity(size.coerceAtMost(128))))
}

/**
 * Returns a [List] containing all elements.
 */
public fun ByteArray.toList(): List<Byte> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun ShortArray.toList(): List<Short> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun IntArray.toList(): List<Int> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun LongArray.toList(): List<Long> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun FloatArray.toList(): List<Float> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun DoubleArray.toList(): List<Double> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun BooleanArray.toList(): List<Boolean> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun CharArray.toList(): List<Char> {
    return when (size) {
        0 -> emptyList()
        else -> this.toMutableList().toList()
    }
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    return ArrayList(this.asCollection())
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun ShortArray.toMutableList(): MutableList<Short> {
    val list = ArrayList<Short>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun IntArray.toMutableList(): MutableList<Int> {
    val list = ArrayList<Int>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun LongArray.toMutableList(): MutableList<Long> {
    val list = ArrayList<Long>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun FloatArray.toMutableList(): MutableList<Float> {
    val list = ArrayList<Float>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun DoubleArray.toMutableList(): MutableList<Double> {
    val list = ArrayList<Double>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun BooleanArray.toMutableList(): MutableList<Boolean> {
    val list = ArrayList<Boolean>(size)
    for (item in this) list.add(item)
    return list
}

/**
 * Returns a new [MutableList] filled with all elements of this array.
 */
public fun CharArray.toMutableList(): MutableList<Char> {
    val list = ArrayList<Char>(size.coerceAtMost(128))
    for (item in this) list.add(item)
    return list
}
[MutableList] filled with all elements of this array.

```java
public fun BooleanArray.toMutableList():
    val list = ArrayList<Boolean>(size)
    for (item in this) list.add(item)
    return list
```

The returned set preserves the element iteration order of the original array.

```java
public fun CharArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Char>(mapCapacity(size))).run {
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun <T> Array<out T>.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<T>(mapCapacity(size))).run {
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun ByteArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Byte>(mapCapacity(size))).run {
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun ShortArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Short>(mapCapacity(size))).run {
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun IntArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Int>(mapCapacity(size))).run {
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun LongArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Long>(mapCapacity(size))).run { 
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun FloatArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Float>(mapCapacity(size))).run { 
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun DoubleArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Double>(mapCapacity(size))).run { 
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun BooleanArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Boolean>(mapCapacity(size))).run { 
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public fun CharArray.toSet():
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Char>(mapCapacity(size.coerceAtMost(128)))).run { 
                run { Returns a [Set] of all elements.
            * The returned set preserves the element iteration order of the original array.
            */
        }
    }
```

```java
public inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```

```java
public inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
```
(Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    Array<out T>.flatMap(transform: (T) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    ByteArray.flatMapIndexed(transform: (index: Int, Byte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    ShortArray.flatMapIndexed(transform: (index: Int, Short) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    IntArray.flatMapIndexed(transform: (index: Int, Int) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> Array<out T>.flatMap(transform: (T) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMap\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> Array<out T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> ByteArray.flatMapIndexed(transform: (index: Int, Byte) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> ShortArray.flatMapIndexed(transform: (index: Int, Short) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    @SinceKotlin("1.4")\n    @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n    @OverloadResolution\n    ByLambdaReturnType\n    @kotlin.jvm.JvmName("flatMapSequence")\n    public inline fun <R> IntArray.flatMapIndexed(transform: (index: Int, Int) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)\n    * Returns a single list of all elements yielded from results of [transform] function being invoked on each element\n    * and its index in the original array.\n    * \n    * @sample samples.collections.Collections.Transformations.flatMapIndexed\n\n    Open Source Used In DNAC Platform Arica  9456
Open Source Used In DNAC Platform Arica 9457
```
results of [transform] function being invoked on each element\n* and its index in the original array, to the given
[destination]\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> ShortArray.flatMapIndexedTo(destination: C, transform: (index: Int,\nShort) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> IntArray.flatMapIndexedTo(destination: C, transform: (index: Int,\nInt) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> LongArray.flatMapIndexedTo(destination: C, transform: (index: Int,\nLong) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> FloatArray.flatMapIndexedTo(destination: C, transform: (index: Int,\nFloat) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> DoubleArray.flatMapIndexedTo(destination: C, transform: (index:\nDouble) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> BooleanArray.flatMapIndexedTo(destination: C, transform: (index:\nBoolean) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}

/**\n * Appends all elements yielded from results of [transform] function being invoked on each element\n* and its index in the original array, to the given\n[destination].\n
/*\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic\nc inline fun <R, C : MutableCollection<in R>> CharArray.flatMapIndexedTo(destination: C, transform: (index:\nChar) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)\n        destination.addAll(list)\n    }\n    return destination\n}
```

---

Open Source Used in DNAC Platform Arica 9458
element
\n    destination.addAll(list)\n\n    return destination\n\n}/*appends all elements yielded from results of \[transform\] function being invoked on each element in this, to the given \[destination\].*/

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].

 * @param destination the destination collection
 * @param transform the transform function
 */
public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given \[keySelector\]

 * @param keySelector the key selector function
 */
public inline fun <T, R, C : MutableCollection<in R>> CharArray.flatMapTo(destination: C, keySelector: (Char) -> R): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
function

* applied to each element and returns a map where each group key is associated with a list of corresponding elements.* The returned map preserves the entry iteration order of the keys produced from the original array.* The returned map preserves the entry iteration order of the keys produced from the original array.* The returned map preserves the entry iteration order of the keys produced from the original array.

```kotlin
public inline fun <T, K> Array<out T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.* The returned map preserves the entry iteration order of the keys produced from the original array.

```kotlin
ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>> {
    return groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)
}

```kotlin
ShortArray.groupBy(keySelector: (Short) -> K): Map<K, List<Short>> {
    return groupByTo(LinkedHashMap<K, MutableList<Short>>(), keySelector)
}

```kotlin
IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>> {
    return groupByTo(LinkedHashMap<K, MutableList<Int>>(), keySelector)
}

```kotlin
LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {
    return groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)
}

```kotlin
FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>> {
    return groupByTo(LinkedHashMap<K, MutableList<Float>>(), keySelector)
}

```kotlin
DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>> {
    return groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)
}

```kotlin
BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>> {
    return groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)
}
CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {
    return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)!
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the
 * element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <T, K, V>
    Array<out T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <K, V>
    ByteArray.groupBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <K, V>
    ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <K, V>
    IntArray.groupBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <K, V>
    LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied
to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
The returned map preserves the entry iteration order of the keys produced from the original array.
 */
public inline fun <K, V>
    FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>>

\{\n  return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n\}

\*\*\*\n
Groups values returned by the [valueTransform] function applied to each element of the original array

\* The returned map preserves the entry iteration order of the keys produced from the original array.

samples.collections.Collections.Transformations.groupByKeysAndValues\n
BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>>

\{\n  return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n\}

\*\*\*\n
Groups values returned by the [valueTransform] function applied to each element of the original array

\* The returned map preserves the entry iteration order of the keys produced from the original array.

CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>>

\{\n  return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n\}

\*\*\*\n
Groups elements of the original array by the key returned by the given [keySelector] function

\* The returned map preserves the entry iteration order of the keys produced from the original array.

samples.collections.Collections.Transformations.groupByKeysAndValues\n
CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>>

\{\n  return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n\}

\*\*\*\n
Groups elements of the original array by the key returned by the given [keySelector] function

\* The returned map preserves the entry iteration order of the keys produced from the original array.

samples.collections.Collections.Transformations.groupByKeysAndValues\n
CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>>

\{\n  return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n\}

\*\*\*\n
Groups elements of the original array by the key returned by the given [keySelector] function

\* The returned map preserves the entry iteration order of the keys produced from the original array.
@return The [destination] map.

@sample samples.collections.Collections.Transformations.groupBy

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Float>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Double>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Boolean>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Char>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Char>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M :MutableMap<in K, MutableList<V>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, V, M :MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: 
(Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
fun Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a list containing the results of applying
 * the given [transform] function to each element in the original array.
 * @return A list containing the results of applying
 * the given [transform] function to each element in the original array.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> LongArray.groupByTo(destination: M, keySelector: 
(Long) -> K, valueTransform: (Long) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableList<V>> FloatArray.groupByTo(destination: M, keySelector: 
(Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableList<V>> DoubleArray.groupByTo(destination: M, keySelector: 
(Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableList<V>> BooleanArray.groupByTo(destination: M, keySelector: 
(Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableList<V>> CharArray.groupByTo(destination: M, keySelector: 
(Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> CharArray.groupByTo(destination: M, keySelector: 
(Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> BooleanArray.groupByTo(destination: M, keySelector: 
(Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> DoubleArray.groupByTo(destination: M, keySelector: 
(Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> FloatArray.groupByTo(destination: M, keySelector: 
(Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> IntArray.groupByTo(destination: M, keySelector: 
(Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> CharArray.groupByTo(destination: M, keySelector: 
(Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> BooleanArray.groupByTo(destination: M, keySelector: 
(Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> DoubleArray.groupByTo(destination: M, keySelector: 
(Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> FloatArray.groupByTo(destination: M, keySelector: 
(Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations on
 * using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Collections.Transformations.map
 */
public inline fun <K, V, M : MutableList<V>> IntArray.groupByTo(destination: M, keySelector: 
(Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}
results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> ByteArray.map(transform: (Byte)\n-> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> ShortArray.map(transform:\n(Short) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> IntArray.map(transform:\n(ToInt) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> LongArray.map(transform:\n(Long) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> FloatArray.map(transform:\n(Float) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> DoubleArray.map(transform:\n(Double) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> BooleanArray.map(transform:\n(Boolean) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element in the original array.\n* \n* \n* @sample
samples.collections.Collections.Transformations.map\n*\npublic inline fun <R> CharArray.map(transform:\n(Char) -> R): List<R> \n{ \n    return mapTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <T, R> Array<out
T>.mapIndexed(transform: (index: Int, T) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> ByteArray.mapIndexed(transform: (index: Int, Byte) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> ShortArray.mapIndexed(transform: (index: Int, Short) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> IntArray.mapIndexed(transform: (index: Int, Int) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> LongArray.mapIndexed(transform: (index: Int, Long) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> FloatArray.mapIndexed(transform: (index: Int, Float) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> DoubleArray.mapIndexed(transform: (index: Int, Double) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> BooleanArray.mapIndexed(transform: (index: Int, Boolean) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
arrays.Collections.Transformations.map\n*\npublic inline fun <R> CharArray.mapIndexed(transform: (index: Int, Char) -> R): List<R> \n{ \n    return mapIndexedTo(ArrayList<R>(size), transform)\n}\n* Returns a list containing the results of applying the given \[\text{transform}\] function\n* to each element and its index in the original array.\n* \n* \n* @param \[\text{transform}\] function that takes the index of an element and the element itself\n* and returns the result of the transform applied to the element.\n* \n* \n* @sample
> R): List<R> { return mapIndexedTo(ArrayList<R>(size), transform) }\n
* Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.\n
* @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.\n
* @public inline fun <Double> DoubleArray.mapIndexedTo(transform: (Int, Double) -> R): List<R> { return mapIndexedTo(ArrayList<R>(size), transform) }\n
* Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.\n
* @public inline fun <Boolean> BooleanArray.mapIndexedTo(transform: (Int, Boolean) -> R): List<R> { return mapIndexedTo(ArrayList<R>(size), transform) }\n
* Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.\n
* @public inline fun <Char> CharArray.mapIndexedTo(transform: (Int, Char) -> R): List<R> { return mapIndexedTo(ArrayList<R>(size), transform) }\n
* Applies the given [transform] function to each element and its index in the original array and appends only the non-null results of applying the given [transform] function to each element and its index in the original array.\n
* @public inline fun <T, R : Any> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (Int, T) -> R?): C { \n    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> FloatArray.mapIndexedTo(destination: C, transform: (Int, Float) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> DoubleArray.mapIndexedTo(destination: C, transform: (Int, Double) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> BooleanArray.mapIndexedTo(destination: C, transform: (Int, Boolean) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> CharArray.mapIndexedTo(destination: C, transform: (Int, Char) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> IntArray.mapIndexedTo(destination: C, transform: (Int, Int) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> LongArray.mapIndexedTo(destination: C, transform: (Int, Long) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> FloatArray.mapIndexedTo(destination: C, transform: (Int, Float) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> DoubleArray.mapIndexedTo(destination: C, transform: (Int, Double) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> BooleanArray.mapIndexedTo(destination: C, transform: (Int, Boolean) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> CharArray.mapIndexedTo(destination: C, transform: (Int, Char) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> IntArray.mapIndexedTo(destination: C, transform: (Int, Int) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> LongArray.mapIndexedTo(destination: C, transform: (Int, Long) -> R): C {
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
 */
public inline fun <R, C : MutableCollection<R>> FloatArray.mapIndexedTo(destination: C, transform: (Int, Float) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of the transform applied to the element.
inline fun <R, C : MutableCollection<in R>> DoubleArray.mapTo(destination: C, transform: (Double) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> BooleanArray.mapTo(destination: C, transform: (Boolean) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> CharArray.mapTo(destination: C, transform: (Char) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element itself.
 */
public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a list containing only distinct elements from the given array.
 * Among equal elements of the given array, only the first one will be present in the resulting list.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <T> Array<out T>.distinctAndDistinctBy(transform: (T) -> Any): List<T> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun ByteArray.distinctAndDistinctBy(transform: (Byte) -> Any): List<Byte> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun ShortArray.distinctAndDistinctBy(transform: (Short) -> Any): List<Short> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun IntArray.distinctAndDistinctBy(transform: (Int) -> Any): List<Int> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun LongArray.distinctAndDistinctBy(transform: (Long) -> Any): List<Long> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun FloatArray.distinctAndDistinctBy(transform: (Float) -> Any): List<Float> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun DoubleArray.distinctAndDistinctBy(transform: (Double) -> Any): List<Double> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun BooleanArray.distinctAndDistinctBy(transform: (Boolean) -> Any): List<Boolean> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun CharArray.distinctAndDistinctBy(transform: (Char) -> Any): List<Char> {
    return this.toMutableSet().toList()
}
public fun LongArray.distinct(): List<Long> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun sample samples.collections.Collections.Transformations.distinctAndDistinctBy

public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only distinct elements from the given array.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList()
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given 
 * [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <T, K> Array<T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given 
 * [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <K> ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {
    val set = HashSet<K>()
    val list = ArrayList<Byte>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given 
 * [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <K> ShortArray.distinctBy(selector: (Short) -> K): List<Short> {
    val set = HashSet<K>()
    val list = ArrayList<Short>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given 
 * [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <K> IntArray.distinctBy(selector: (Int) -> K): List<Int> {
    val set = HashSet<K>()
    val list = ArrayList<Int>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given 
 * [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public fun <K> LongArray.distinctBy(selector: (Long) -> K): List<Long> {
    val set = HashSet<K>()
    val list = ArrayList<Long>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
FloatArray.distinctBy(selector: (Float) -> K): List<Float> {
    val set = HashSet<K>()
    val list = ArrayList<Float>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public inline fun <K> DoubleArray.distinctBy(selector: (Double) -> K): List<Double> {
    val set = HashSet<K>()
    val list = ArrayList<Double>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public inline fun <K> BooleanArray.distinctBy(selector: (Boolean) -> K): List<Boolean> {
    val set = HashSet<K>()
    val list = ArrayList<Boolean>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
public inline fun <K> CharArray.distinctBy(selector: (Char) -> K): List<Char> {
    val set = HashSet<K>()
    val list = ArrayList<Char>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 * The elements in the resulting set are in the same order as they were in the source array.
 */
public infix fun <T> Array<out T>.intersect(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun ShortArray.intersect(other: Iterable<Short>): Set<Short> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun IntArray.intersect(other: Iterable<Int>): Set<Int> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun LongArray.intersect(other: Iterable<Long>): Set<Long> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun FloatArray.intersect(other: Iterable<Float>): Set<Float> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by both this array and the specified collection.
 * The returned set preserves the element iteration order of the original array.
 */
public infix fun DoubleArray.intersect(other: Iterable<Double>): Set<Double> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}
val set = this.toMutableSet()
    set.retainAll(other)
    return set

Returns a set containing all elements that are contained by both this array and the specified collection. The returned set preserves the element iteration order of the original array. To get a set containing all elements that are contained at least in one of these collections use [union].

public infix fun BooleanArray.intersect(other: Iterable<Boolean>): Set<Boolean> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

Returns a set containing all elements that are contained by this array and not contained by the specified collection. The returned set preserves the element iteration order of the original array.

public infix fun CharArray.intersect(other: Iterable<Char>): Set<Char> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

Returns a new [MutableSet] containing all distinct elements from the given array. The returned set preserves the element iteration order of the original array.

public fun <T> Array<out T>.subtract(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

Returns a new [MutableSet] containing all distinct elements from the given array. The returned set preserves the element iteration order of the original array.
of the original array.

```kotlin
/*/public fun IntArray.toMutableSet(): MutableSet<Int> {
    return toCollection(LinkedHashSet<Int>(mapCapacity(size)))
}*/

/**
 * Returns a new [MutableSet] containing all unique elements from the given array.
 * The returned set preserves the element iteration order of the original array.
 */
public fun LongArray.toMutableSet(): MutableSet<Long> {
    return toCollection(LinkedHashSet<Long>(mapCapacity(size)))
}

/**
 * Returns a new [MutableSet] containing all unique elements from the given array.
 * The returned set preserves the element iteration order of the original array.
 */
public fun FloatArray.toMutableSet(): MutableSet<Float> {
    return toCollection(LinkedHashSet<Float>(mapCapacity(size)))
}

/**
 * Returns a new [MutableSet] containing all unique elements from the given array.
 * The returned set preserves the element iteration order of the original array.
 */
public fun DoubleArray.toMutableSet(): MutableSet<Double> {
    return toCollection(LinkedHashSet<Double>(mapCapacity(size)))
}

/**
 * Returns a new [MutableSet] containing all unique elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the end
 * in the order of the [other] collection.
 * To get a set containing all elements that are contained in both collections use
 * [intersect].
 */
public infix fun <T> Array<out T>.union(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns a set containing all distinct elements from both collections.
 * Those elements of the [other] collection that are unique are iterated in the end
 * in the order of the [other] collection.
 * To get a set containing all elements that are contained in both collections use
 * [intersect].
 */
public infix fun ByteArray.union(other: Iterable<Byte>): Set<Byte> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}
```

**Open Source Used In DNAC Platform Arica**
public infix fun DoubleArray.union(other: Iterable<Double>): Set<Double> {
    val set = this.toDoubleArray().toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns a set containing all distinct elements from both collections.
 * The returned set preserves the element iteration order of the original array.
 * To get a set containing all elements that are contained in both collections use [intersect].
 */
public infix fun BooleanArray.union(other: Iterable<Boolean>): Set<Boolean> {
    val set = this.toDoubleArray().toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns a set containing all distinct elements from both collections.
 * The returned set preserves the element iteration order of the original array.
 * To get a set containing all elements that are contained in both collections use [intersect].
 */
public infix fun CharArray.union(other: Iterable<Char>): Set<Char> {
    val set = this.toDoubleArray().toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun <T> Array<out T>.all(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun ByteArray.all(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun <T> Array<out T>.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun ByteArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun ShortArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun IntArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun LongArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun FloatArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun DoubleArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun BooleanArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun CharArray.any(): Boolean {
    return !isEmpty()
}
samples.collections.Collections.Aggregates.any

```kotlin
public fun DoubleArray.any(): Boolean {
    return
    isEmpty
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun BooleanArray.any(): Boolean {
    return
    isEmpty
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun CharArray.any(): Boolean {
    return
    isEmpty
}

/**
 * Returns `true` if array has at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun <T> Array<out T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun ByteArray.any(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun ShortArray.any(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun IntArray.any(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun LongArray.any(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun FloatArray.any(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.any
 */
public fun CharArray.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements in this array.
 */
public inline

```
Open Source Used In DNAC Platform Arica 9475

with \[initial\] value and applying \[operation\] from left to right to current accumulator value and each element.

* to current accumulator value and each element.

\[
\begin{align*}
\text{FloatArray} &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{Float}) \to \mathbb{R}) : \mathbb{R} \\
\text{LongArray} &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{Long}) \to \mathbb{R}) : \mathbb{R} \\
\text{IntArray} &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{Int}) \to \mathbb{R}) : \mathbb{R} \\
\text{ShortArray} &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{Short}) \to \mathbb{R}) : \mathbb{R} \\
\text{ByteArray} &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{Byte}) \to \mathbb{R}) : \mathbb{R} \\
\text{T}. &. \text{fold}(\text{initial}: \mathbb{R}, \text{operation}: (\text{acc}: \mathbb{R}, \text{T}) \to \mathbb{R}) : \mathbb{R}
\end{align*}
\]

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.

* Returns the specified \[initial\] value if the array is empty.
\n
* Returns the specified [initial] value if the array is empty.

```kotlin
public inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {\n    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

* Returns the specified [initial] value if the array is empty.

```kotlin
public inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {\n    var accumulator = initial\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

* Returns the specified [initial] value if the array is empty.

```kotlin
public inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial\n    return accumulator\n} \n
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
public inline fun <R> ShortArray.fold(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> ByteArray.fold(initial: R, operation: (index: Int, acc: R, Byte) -> R): R {
    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
public inline fun <T, R> Array<out T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)
    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> LongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> ShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial\n    return accumulator\n} \n
* @param [operation] function that takes current accumulator value and each element.

```kotlin
public inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    var accumulator = initial\n    for (element in this) accumulator = operation(accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {
    var accumulator = initial\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.

```kotlin
public inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial\n    return accumulator\n} \n
* @param [operation] function that takes current accumulator value and each element.

```kotlin
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n} \n
* @param [operation] function that takes the index of an element, current accumulator value and each element.
itself, and calculates the next accumulator value.

```kotlin
/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array. 
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 */
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array. 
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 */
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array. 
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 */
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to current accumulator value and each element with its index in the original array. 
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <T, R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array. 
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 */
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
```

Open Source Used In DNAC Platform Arica 9477
accumulate = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> FloatArray.foldRightIndexed(initial: R, operation: (index: Int, Float, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> LongArray.foldRightIndexed(initial: R, operation: (index: Int, Long, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> ByteArray.forEach(action: (Byte) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> IntArray.forEach(action: (Int) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> ShortArray.forEach(action: (Short) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> DoubleArray.forEach(action: (Double) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> FloatArray.forEach(action: (Float) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> IntArray.forEach(action: (Int) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> LongArray.forEach(action: (Long) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <R> ShortArray.forEach(action: (Short) -> Unit): Unit {
    for (element in this) action(element)
}
element.

@param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun <T> Array-out T>-foreachIndexed(action: (index: Int, T) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun ShortArray.foreachIndexed(action: (index: Int, Short) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun IntArray.foreachIndexed(action: (index: Int, Int) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun LongArray.foreachIndexed(action: (index: Int, Long) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun FloatArray.foreachIndexed(action: (index: Int, Float) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun DoubleArray.foreachIndexed(action: (index: Int, Double) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun BooleanArray.foreachIndexed(action: (index: Int, Boolean) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n*\n* Performs the given [action] on each element, providing sequential index with the element.

* @param [action] function that takes the index of an element and the element itself

* and performs the action on the element.

*\n*public inline fun CharArray.foreachIndexed(action: (index: Int, Char) -> Unit):

Unit {\n    var index = 0\n    for (item in this) action(index++, item)\n
@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Array<out Double>.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun Array<out Float>.max(): Float? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Array<out T>.max(): T? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.max(): Byte? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.max(): Short? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.max(): Int? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.max(): Long? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
    ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun FloatArray.max(): Float? {
    return maxOrNull()
}
DoubleArray.max(): Double? {\n    return maxOrNull()\n}\n@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))\npublic fun CharArray.max(): Char? {\n    return maxOrNull()\n}\n@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))\npublic fun DoubleArray.max(): Double? {
    return maxOrNull()
}\n@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))\n@DeprecatedSinceKotlin(warningSince = "1.4")\npublic fun CharArray.maxBy(selector: (Char) -> R): Char? {
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <T, R> Array<out T>.maxByOrNull(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> ByteArray.maxByOrNull(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> ShortArray.maxByOrNull(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> IntArray.maxByOrNull(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> LongArray.maxByOrNull(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> DoubleArray.maxByOrNull(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> BooleanArray.maxByOrNull(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> CharArray.maxByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> ByteArray.maxByOrNull(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> ShortArray.maxByOrNull(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> IntArray.maxByOrNull(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> LongArray.maxByOrNull(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> DoubleArray.maxByOrNull(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> BooleanArray.maxByOrNull(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R> CharArray.maxByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
`null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> IntArray.maxByOrNull(selector: (Int) -> R):
    Int? {
        if (isEmpty()) return null
        var maxElem = this[0]
        val lastIndex = this.lastIndex
        if (lastIndex == 0) return maxElem
        var maxVal = selector(maxElem)
        for (i in 1..lastIndex) {
            val e = this[i]
            val v = selector(e)
            if (maxVal < v) {
                maxElem = e
                maxVal = v
            }
        }
        return maxElem
    }
```

```kotlin
/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> LongArray.maxByOrNull(selector: (Long) ->
    R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxVal = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxVal < v) {
            maxElem = e
            maxVal = v
        }
    }
    return maxElem
}
```

```kotlin
/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> FloatArray.maxByOrNull(selector: (Float) ->
    R): Float? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxVal = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxVal < v) {
            maxElem = e
            maxVal = v
        }
    }
    return maxElem
}
```

```kotlin
/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> DoubleArray.maxByOrNull(selector: (Double) ->
    R): Double? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxVal = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxVal < v) {
            maxElem = e
            maxVal = v
        }
    }
    return maxElem
}
```

```kotlin
/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> BooleanArray.maxByOrNull(selector: (Boolean) -> R):
    Boolean? {
        if (isEmpty()) return null
        var maxElem = this[0]
        val lastIndex = this.lastIndex
        if (lastIndex == 0) return maxElem
        var maxVal = selector(maxElem)
        for (i in 1..lastIndex) {
            val e = this[i]
            val v = selector(e)
            if (maxVal < v) {
                maxElem = e
                maxVal = v
            }
        }
        return maxElem
    }
```

```kotlin
/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.maxOf(selector: (T) -> Double): Double? {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```
```
Double): Double {\n    if (isNotEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.maxOf(selector: (Short) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.maxOf(selector: (Int) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.maxOf(selector: (Long) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.maxOf(selector: (Float) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.maxOf(selector: (Double) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}\n
/**
 * Returns the largest value among all values produced by [selector] function applied to each element in the array.\n * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.\n * @throws NoSuchElementException if the array is empty.\n */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.maxOf(selector: (Boolean) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()\n    var maxValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        maxValue = maxOf(maxValue, v)\n    }\n    return maxValue\n}
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun DoubleArray.maxOf(selector: (Double) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun BooleanArray.maxOf(selector: (Boolean) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun CharArray.maxOf(selector: (Char) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun <T, R : Comparable<R>> Array<out T>.maxOf(selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun <R : Comparable<R>> ByteArray.maxOf(selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun <R : Comparable<R>> ShortArray.maxOf(selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] applied to each element in the array.
 * If any of values produced by [selector] is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@in public inline fun <R : Comparable<R>> IntArray.maxOf(selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/*
 * Open Source Used In DNAC Platform Arica 9485
 */
*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <R : Comparable<R>>
*\@LongArray.maxOf(selector: (Long) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <R : Comparable<R>>
*\@FloatArray.maxOf(selector: (Float) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <R : Comparable<R>>
*\@DoubleArray.maxOf(selector: (Double) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <R : Comparable<R>>
*\@BooleanArray.maxOf(selector: (Boolean) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <R : Comparable<R>>
*\@CharArray.maxOf(selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

*\@Since Kotlin("1.4")\n*\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n*\@OverloadResolution
*\@LambdaReturnType
*\@kotlin.internal.InlineOnly
*\@public inline fun ByteArray.maxOfOrNull(selector: (Byte) -> Double): Double? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
The code snippet is a Kotlin function that finds the maximum value among all values produced by a given selector function applied to each element in an array. The function is defined for different array types: `ShortArray`, `IntArray`, `LongArray`, `FloatArray`, and `DoubleArray`. It uses a lambda expression to apply the selector function to each element of the array and returns the maximum value or `null` if the array is empty. The function checks if the result of any selector function application is `NaN` and returns `NaN` in such cases. The code is structured with Kotlin's function overloading, where the return type and parameter type are specified for each array type.
if there are no elements.\n * \n * If any of values produced by [selector] function is `NaN`, the returned result is 
 * `'NaN'`.\n
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.maxOfOrNull(selector: (T) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun ByteString.maxOfOrNull(selector: (Byte) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun ShortArray.maxOfOrNull(selector: (Short) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun IntArray.maxOfOrNull(selector: (Int) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun LongArray.maxOfOrNull(selector: (Long) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun FloatArray.maxOfOrNull(selector: (Float) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly
public inline fun DoubleArray.maxOfOrNull(selector: (Double) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```
maxValue

/* Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun BooleanArray.maxOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun CharArray.maxOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public <T, R : Comparable<R>> Array<out T>.maxOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun ByteArray.maxOfOrNull(selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun ShortArray.maxOfOrNull(selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun IntArray.maxOfOrNull(selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function.*/

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun LongArray.maxOfOrNull(selector: (Long) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
FloatArray.maxOfOrNull(selector: (Float) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R : Comparable<R>>
DoubleArray.maxOfOrNull(selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R : Comparable<R>>
BooleanArray.maxOfOrNull(selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R : Comparable<R>>
CharArray.maxOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R : Comparable<R>>
Array<out T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R> ByteArray.maxOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R> ShortArray.maxOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced
 * by [selector] function applied to each element in the array
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
-inline fun <R> IntArray.maxOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
= selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <R> LongArray.maxOfWith(comparator: Comparator<R>, selector: (Long) -> R): R {
     if (isEmpty()) throw NoSuchElementException()
     var maxValue = selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     }

     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <R> FloatArray.maxOfWith(comparator: Comparator<R>, selector: (Float) -> R): R {
     if (isEmpty()) throw NoSuchElementException()
     var maxValue = selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     }

     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <R> DoubleArray.maxOfWith(comparator: Comparator<R>, selector: (Double) -> R): R {
     if (isEmpty()) throw NoSuchElementException()
     var maxValue = selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     }

     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <R> BooleanArray.maxOfWith(comparator: Comparator<R>, selector: (Boolean) -> R): R {
     if (isEmpty()) throw NoSuchElementException()
     var maxValue = selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     }

     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <T, R> Array<out T>.maxOfWithOrNull(comparator: Comparator<R>, selector: (T) -> R): R? {
     if (isEmpty()) return null
     var maxValue = selector(this[0])
     for (i in 1..lastIndex) { val v = selector(this[i])
     if (comparator.compare(maxValue, v) < 0) { maxValue = v
     } return maxValue
     }
     return maxValue
     }

     *\n
     Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.\n
     *\n
     @Throws NoSuchElementException if the array is empty.\n
     ASTM@SinceKotlin("1.4")\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n
     *\n
     ByLambdaReturnType\n
     @OptIn(kotlin.internal.InlineOnly)\n
     *\n
     @public inline fun <R>
ByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? { 
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector]
 * function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Boolean) -> R): R? {
    if (isNullOrEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
ByLambdaReturnType

@kotlin.internal.InlineOnly
public inline fun <R>
CharArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R)? {
    @n if (isEmpty()) return null
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxVal, v) < 0) {
            maxVal = v
        }
    }
    return maxVal
}

@SinceKotlin("1.4")
public inline fun <R>
CharArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) {
            max = e
        }
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun Array<out Double>.maxOrNull(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun Array<out Float>.maxOrNull(): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun Array<out T : Comparable<T>>.maxOrNull(): T? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun ByteArray.maxOrNull(): Byte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun ShortArray.maxOrNull(): Short? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun IntArray.maxOrNull(): Int? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun LongArray.maxOrNull(): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@SinceKotlin("1.4")
public fun FloatArray.maxOrNull(): Float? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        max = maxOf(max, e)
    }
    return max
}

@SinceKotlin("1.4")
public fun DoubleArray.maxOrNull(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        max = maxOf(max, e)
    }
    return max
}

@SinceKotlin("1.4")
public fun CharArray.maxOrNull(): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Array<out T>.maxWith(comparator: Comparator<in T>): T? {
    @n if (isEmpty()) return null
    return maxWithOrNull(comparator)
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.maxWith(comparator: Comparator<in Byte>): Byte? {
    @n if (isEmpty()) return null
    return maxWithOrNull(comparator)
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.maxWith(comparator: Comparator<in Short>): Short? {
    @n if (isEmpty()) return null
    return maxWithOrNull(comparator)
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.maxWith(comparator: Comparator<in Int>): Int? {
    @n if (isEmpty()) return null
    return maxWithOrNull(comparator)
}

@Deprecated("Use maxWithOrNull instead.",
    ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.maxWith(comparator: Comparator<in Long>): Long? {
    @n if (isEmpty()) return null
    return maxWithOrNull(comparator)
}
Open Source Used In DNAC Platform Arica
ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun Array<out Double>.min(): Double? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun Array<out Float>.min(): Float? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun Array<out T : Comparable<T>>.min(): T? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun ByteArray.min(): Byte? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun ShortArray.min(): Short? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun IntArray.min(): Int? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun LongArray.min(): Long? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun FloatArray.min(): Float? {
    return minOrNull()
}

ReplaceWith("minOrNull()")@DeprecatedSinceKotlin(warningSince = "1.4")@SinceKotlin("1.1")@public fun DoubleArray.min(): Double? {
    return minOrNull()
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")@public inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> ByteArray.minBy(selector: (Byte) -> R): Byte? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> ShortArray.minBy(selector: (Short) -> R): Short? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> DoubleArray.minBy(selector: (Double) -> R): Double? {
    return minByOrNull(selector)
}

@SinceKotlin("1.4")@public inline fun <R : Comparable<R>> BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {
    return minByOrNull(selector)
}

* Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
fun <T, R : Comparable<R>> Array<out T>.minByOrNull(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
        println(minElem) // Assuming println is defined somewhere
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> BooleanArray.minByOrNull(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> DoubleArray.minByOrNull(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> FloatArray.minByOrNull(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> IntArray.minByOrNull(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> LongArray.minByOrNull(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> ShortArray.minByOrNull(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samplescollections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minByOrNull
 */
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (CharSequence) -> R): CharSequence? {
    if (isEmpty()) return null
    var minElem = this
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

fun <T> Array<out T>.minByOrNull(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElement = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElement
    var minVal = selector(minElement)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minVal > v) {
            minElement = e
            minVal = v
        }
        println(minElement) // Assuming println is defined somewhere
    }
    return minElement
}

fun <T, R : Comparable<R>> Array<out T>.minByOrNull(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElement = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElement
    var minVal = selector(minElement)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minVal > v) {
            minElement = e
            minVal = v
        }
    }
    return minElement
}
<R : Comparable<R>> CharArray.minByOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.minOf(selector: (T) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun ByteArray.minOf(selector: (Byte) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun ShortArray.minOf(selector: (Short) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun IntArray.minOf(selector: (Int) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun LongArray.minOf(selector: (Long) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun FloatArray.minOf(selector: (Float) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun DoubleArray.minOf(selector: (Double) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun BooleanArray.minOf(selector: (Boolean) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun CharArray.minOf(selector: (Char) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun FloatArray.minOf(selector: (Float) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ByteArray.minOf(selector: (Byte) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ShortArray.minOf(selector: (Short) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.minOf(selector: (Int) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ShortArray.minOf(selector: (Short) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the array.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.minOf(selector: (Int) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
each element in the array. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is ` NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty. \n
If any of values produced by [selector] function is `NaN`, the returned result is `NaN`. \n
Throws NoSuchElementException if the array is empty.
by [selector] function\n* applied to each element in the array.\n* \n* @throws NoSuchElementException if the array is empty.\n
```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
ShortArray.minOf(selector: (Short) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
IntArray.minOf(selector: (Int) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
LongArray.minOf(selector: (Long) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
FloatArray.minOf(selector: (Float) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
DoubleArray.minOf(selector: (Double) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType\n@kotlin.internal.InlineOnly\npublic inline fun <R : Comparable<R>>
CharArray.minOf(selector: (Char) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}  // Returns the smallest value among all values

```

Open Source Used In DNAC Platform Arica 9500
** @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
def minOfOrNull<T>(selector: (T) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun <T> Array<out T>.minOfOrNull(selector: (T) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun ByteArray.minOfOrNull(selector: (Byte) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun ShortArray.minOfOrNull(selector: (Short) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun IntArray.minOfOrNull(selector: (Int) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun LongArray.minOfOrNull(selector: (Long) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun FloatArray.minOfOrNull(selector: (Float) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution ByLambdaReturnType @kotlin.internal.InlineOnly
fun DoubleArray.minOfOrNull(selector: (Double) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```kotlin
1..lastIndex) {
    val v = selector(this[i])
    minValue = minOf(minValue, v)
} return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun CharArray.minOfOrNull(selector: (Char) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns
 * the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.minOfOrNull(selector: (T) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ByteArray.minOfOrNull(selector: (Byte) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun ShortArray.minOfOrNull(selector: (Short) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun IntArray.minOfOrNull(selector: (Int) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

Open Source Used In DNAC Platform Arica 9502
> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public fun FloatArray.minOfOrNull(selector: (Float) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.internal.InlineOnly::class)
public inline fun DoubleArray.minOfOrNull(selector: (Double) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.internal.InlineOnly::class)
public inline fun BooleanArray.minOfOrNull(selector: (Boolean) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
public inline fun CharArray.minOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.util.InlineOnly::class)
public inline fun <T, R : Comparable<R>> Array<out T>.minOfOrNull(selector: (T) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.util.InlineOnly::class)
public inline fun <R : Comparable<R>> ByteArray.minOfOrNull(selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.util.InlineOnly::class)
public inline fun <R : Comparable<R>> ShortArray.minOfOrNull(selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<IntArray>.minOfOrNull(selector: (Int) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<LongArray>.minOfOrNull(selector: (Long) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<FloatArray>.minOfOrNull(selector: (Float) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<DoubleArray>.minOfOrNull(selector: (Double) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<CharArray>.minOfOrNull(selector: (Char) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<CharArray>.minOfOrNull(selector: (Char) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function whenever applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
public inline fun<CharArray>.minOfOrNull(selector: (Char) -> R?): R? {
  if (isEmpty()) return null
  var minValue = selector(this[0])
  for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (minValue > v) {
      minValue = v
    }
  }
  return minValue
}
ByLambdaReturnType

```kotlin
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.minOfWith(comparator: Comparator<in R>, selector: (Byte) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWith(comparator: Comparator<in R>, selector: (Short) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.minOfWith(comparator: Comparator<in R>, selector: (Int) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.minOfWith(comparator: Comparator<in R>, selector: (Long) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.minOfWith(comparator: Comparator<in R>, selector: (Float) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.minOfWith(comparator: Comparator<in R>, selector: (Double) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.minOfWith(comparator: Comparator<in R>, selector: (Boolean) -> R): R {
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

- **Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.**
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.minOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Byte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Short) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Int) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Long) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Float) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
 */
function applied to each element in the array or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <R>

DoubleArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Double) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
BooleanArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Boolean) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
CharArray.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
Array<out Double>.minOrNull(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}
```

```kotlin
Array<out Float>.minOrNull(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}
```

```kotlin
Array<out T : Comparable<T>>.minOrNull(): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

```kotlin
ByteArray.minOrNull(): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

```kotlin
ShortArray.minOrNull(): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

```kotlin
IntArray.minOrNull(): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

```kotlin
LongArray.minOrNull(): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

```kotlin
FloatArray.minOrNull(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}
```

```kotlin
DoubleArray.minOrNull(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        min = minOf(min, e)
    }
    return min
}
```

```kotlin
CharArray.minOrNull(): Char? {
    if (isEmpty()) return null
    var min =
for (i in 1..lastIndex) { 
    val e = this[i]
    if (min > e) min = e
} 
return min
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun FloatArray.minWith(comparator: Comparator<in Float>): Float? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {
    return minWithOrNull(comparator)
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    return minWithOrNull(comparator)
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun <T> Array<out T>.minWithOrNull(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ByteArray.minWithOrNull(comparator: Comparator<in Byte>): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun ShortArray.minWithOrNull(comparator: Comparator<in Short>): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun IntArray.minWithOrNull(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun LongArray.minWithOrNull(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun FloatArray.minWithOrNull(comparator: Comparator<in Float>): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun DoubleArray.minWithOrNull(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun BooleanArray.minWithOrNull(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided 
 * comparator or `null` if there are no elements.
 */
@SinceKotlin("1.4")
public fun CharArray.minWithOrNull(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
smaller value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
public fun FloatArray.minWithOrNull(comparator: Comparator<in Float>): Float? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@SinceKotlin("1.4")
public fun DoubleArray.minWithOrNull(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@SinceKotlin("1.4")
public fun BooleanArray.minWithOrNull(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@SinceKotlin("1.4")
public fun CharArray.minWithOrNull(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@SinceKotlin("1.4")
public fun ByteArray.minWithOrNull(comparator: Comparator<in Byte>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

@SinceKotlin("1.4")
public fun ShortArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun IntArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun LongArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun FloatArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun DoubleArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun BooleanArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

@SinceKotlin("1.4")
public fun CharArray.minWithOrNull(): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```
Returns `true` if no elements match the given [predicate].

```kotlin
public inline fun FloatArray.none(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

Returns `true` if no elements match the given [predicate].

```kotlin
public inline fun DoubleArray.none(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

Returns `true` if no elements match the given [predicate].

```kotlin
public inline fun BooleanArray.none(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

Returns `true` if no elements match the given [predicate].

```kotlin
public inline fun CharArray.none(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.onEach(action: (T) -> Unit): Array<out T> {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEach(action: (Byte) -> Unit): ByteArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.onEach(action: (Short) -> Unit): ShortArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.onEach(action: (Int) -> Unit): IntArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun LongArray.onEach(action: (Long) -> Unit): LongArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.onEach(action: (Float) -> Unit): FloatArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.onEach(action: (Double) -> Unit): DoubleArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEach(action: (Boolean) -> Unit): BooleanArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEach(action: (Char) -> Unit): CharArray {
    return apply { for (element in this) action(element) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Unit): ByteArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.onEachIndexed(action: (index: Int, Short) -> Unit): ShortArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.onEachIndexed(action: (index: Int, Int) -> Unit): IntArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun LongArray.onEachIndexed(action: (index: Int, Long) -> Unit): LongArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.onEachIndexed(action: (index: Int, Float) -> Unit): FloatArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.onEachIndexed(action: (index: Int, Double) -> Unit): DoubleArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

Performs the given [action] on each element, providing sequential index with the element, and returns the array itself afterwards.
with the element.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ShortArray.onEachIndexed(action: (index: Int, Short) -> Unit): ShortArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.onEachIndexed(action: (index: Int, Int) -> Unit): IntArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun LongArray.onEachIndexed(action: (index: Int, Long) -> Unit): LongArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.onEachIndexed(action: (index: Int, Float) -> Unit): FloatArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.onEachIndexed(action: (index: Int, Double) -> Unit): DoubleArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun ByteArray.onEachIndexed(action: (index: Int, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.onEachIndexed(action: (index: Int, Boolean) -> Unit): BooleanArray {
    return apply { forEachIndexed(action) }
}
```

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.onEachIndexed(action: (index: Int, Char) -> Unit): CharArray {
    return apply { forEachIndexed(action) }
}
```
in an expected way,\n* please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**ShortArray.reduce(operation: (acc: Short, Short) -> Short): Short\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**IntArray.reduce(operation: (acc: Int, Int) -> Int): Int\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**LongArray.reduce(operation: (acc: Long, Long) -> Long): Long\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**FloatArray.reduce(operation: (acc: Float, Float) -> Float): Float\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**DoubleArray.reduce(operation: (acc: Double, Double) -> Double): Double\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce\n
**BooleanArray.reduce(operation: (acc: Boolean, Boolean) -> Boolean): Boolean\n* if (isEmpty())\n* throw UnsupportedOperationException("Empty array can't be reduced.")\n* var accumulator = this[0]\n* for (index in 1..lastIndex) {\n* accumulator = operation(accumulator, this[index])\n* }\n* return accumulator\n**

Accumulates value starting with the first element and applying [operation] from left to right\n* to current accumulator value and each element.\n* @param [operation] function that takes current accumulator value and an element,\n* and calculates the next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce
in an expected way.\n * please use [reduceOrNull] instead. It returns ‘null’ when its receiver is empty.\n * \n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n
public inline fun CharArray.reduce(operation: (acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param [operation] function that takes the index of an element, current\n * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun <T, S> Array<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator: S = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param [operation] function that takes the index of an element, current\n * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param [operation] function that takes the index of an element, current\n * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param [operation] function that takes the index of an element, current\n * accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {  
        accumulator = operation(index, accumulator, this[index])
    }  
    return accumulator
}

@sample samples.collections.Collections.Aggregates.reduce
public inline fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@sample samples.collections.Collections.Aggregates.reduce
public inline fun DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@sample samples.collections.Collections.Aggregates.reduce
public inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@sample samples.collections.Collections.Aggregates.reduce
public inline fun CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@sample samples.collections.Collections.Aggregates.reduce
public inline fun <S, T : S> Array<out T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S): S? {
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
in the original array.\n * \n * Returns `null` if the array is empty.\n * \n * @param [operation] function that takes the\n * index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
public inline fun ByteArray.reduceIndexedOrNull(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte? {
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}\n\n/**\n * Accumulates value starting with the first element and applying \[operation\] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param \[operation\] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
public inline fun ShortArray.reduceIndexedOrNull(operation: (index: Int, acc: Short, Short) -> Short): Short? {
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}\n\n/**\n * Accumulates value starting with the first element and applying \[operation\] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param \[operation\] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
public inline fun IntArray.reduceIndexedOrNull(operation: (index: Int, acc: Int, Int) -> Int): Int? {
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}\n\n/**\n * Accumulates value starting with the first element and applying \[operation\] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param \[operation\] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}\n\n/**\n * Accumulates value starting with the first element and applying \[operation\] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param \[operation\] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
public inline fun FloatArray.reduceIndexedOrNull(operation: (index: Int, acc: Float, Float) -> Float): Float? {
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}\n\n/**\n * Accumulates value starting with the first element and applying \[operation\] from left to right\n * to current accumulator value and each element with its index in the original array.\n * \n * @param \[operation\] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.\n * \n * @sample \n * \n * @sample samples.collections.Collections.Aggregates.reduceOrNull\n */
@SinceKotlin("1.4")
public inline fun DoubleArray.reduceIndexedOrNull(operation: (index: Int, acc: Double, Double) -> Double): Double? {
    \n    \n    var accumulator = this[0]
    \n    return accumulator
}
BooleanArray.reduceIndexedOrNull(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean? {
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself.
 * @returns the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceIndexedOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <S, T : S> Array<S>.reduceIndexedOrNull(operation: (acc: S, T) -> S): S? {
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param [operation] function that takes current accumulator value and an element.
 * @returns the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceIndexedOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun Array<Byte>.reduceIndexedOrNull(operation: (acc: Byte, Byte) -> Byte): Byte? {
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param [operation] function that takes current accumulator value and an element.
 * @returns the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceIndexedOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun Array<Int>.reduceIndexedOrNull(operation: (acc: Int, Int) -> Int): Int? {
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * @param [operation] function that takes current accumulator value and an element.
 * @returns the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceIndexedOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    if (isEmpty()) return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
accumulator.value

@sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull

*\n @sample samples.collections.Collections.Aggregates.reduceOrNull
and calculates the next accumulator value.

```kotlin
// * @sample
samples.collections.Collections.Aggregates.reduceRight

/*
 * Calculates value starting with the last element and applying [operation] from right to left to each element
 * and current accumulator value. * * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
 * * @param [operation] function that takes an element and current accumulator value,
left\n* to each element and current accumulator value.\n* \n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n* \n* please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.\n* \n* @param [operation] function that takes an element and current accumulator value,\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull\n*public inline fun CharArray.reduceRight(operation:\n* (Char, acc: Char) -> Char): Char {
*     var index = lastIndex\n*     if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n*     var accumulator = get(index--)\n*     while (index >= 0) {\n*         accumulator = operation(get(index--), accumulator)\n*         return accumulator\n*     }\n*     return accumulator\n* }

/**
* Accumulates value starting with the last element and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightIndexed\n*/
public inline fun <S, T : S> Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {
*     var index = lastIndex\n*     if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n*     var accumulator: S = get(index--)\n*     while (index >= 0) {\n*         accumulator = operation(index, get(index), accumulator)\n*         --index\n*         }\n*         return accumulator\n*     }

/**
* Accumulates value starting with the last element and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightIndexed\n*/
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
*     var index = lastIndex\n*     if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n*     var accumulator = get(index--)\n*     while (index >= 0) {\n*         accumulator = operation(index, get(index), accumulator)\n*         --index\n*         }\n*         return accumulator\n*     }

/**
* Accumulates value starting with the last element and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightIndexed\n*/
public inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {
*     var index = lastIndex\n*     if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n*     var accumulator = get(index--)\n*     while (index >= 0) {\n*         accumulator = operation(index, get(index), accumulator)\n*         --index\n*         }\n*         return accumulator\n*     }

/**
* Accumulates value starting with the last element and applying [operation] from right to left\n* to each element with its index in the original array and current accumulator value.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value.\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightIndexed\n*/
public inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int {
*     var index = lastIndex\n*     if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")\n*     var accumulator = get(index--)\n*     while (index >= 0) {\n*         accumulator = operation(index, get(index), accumulator)\n*         --index\n*         }\n*         return accumulator\n*     }\n

exception if this array is empty. If the array can be empty in an expected way,\n* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty,\n* and calculates the next accumulator value.\n* and calculates the next accumulator value.\n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* and calculates the next accumulator value.\n* * * @sample samples.collections.Collections.Aggregates.reduceRight

```kotlin
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying \[operation\] from right to left\n* to each element with its index in the original array and current accumulator value.\n* * @param \[operation\] function that takes the index of an element, the element itself and current accumulator value,\n* * and calculates the next accumulator value.\n* * * @sample samples.collections.Collections.Aggregates.reduceRight

```kotlin
```
from right to left\n* to each element with its index in the original array and current accumulator value.\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun <S, T : S> Array<out T>.reduceRightIndexedOrNull(operation: (index: Int, T, acc: S) -> S): S? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator: S = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun ByteArray.reduceRightIndexedOrNull(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun ShortArray.reduceRightIndexedOrNull(operation: (index: Int, Short, acc: Short) -> Short): Short? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun IntArray.reduceRightIndexedOrNull(operation: (index: Int, Int, acc: Int) -> Int): Int? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun LongArray.reduceRightIndexedOrNull(operation: (index: Int, Long, acc: Long) -> Long): Long? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
@SinceKotlin("1.4")
public inline fun FloatArray.reduceRightIndexedOrNull(operation: (index: Int, Float, acc: Float) -> Float): Float? {\n    var index = lastIndex\n    if (index < 0) return null\n    var accumulator = get(index--)\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n        }\n    return accumulator\n}\n* \n* Returns `null` if the array is empty.\n* \n* @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n* \n* and calculates the next accumulator value,\n* \n* @sample samples.collections.Collections.Aggregates.reduceRightOrNull*/
Accumulates value starting with the last element and applying [operation] from right to left on each element with its index in the original array and current accumulator value.

@returns `null` if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.reduceRightOrNull
*/
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun ByteArray.reduceRightOrNull(operation: (Byte, acc: Byte) -> Byte): Byte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/*
Accumulates value starting with the last element and applying [operation] from right to left on each element with its index in the original array and current accumulator value.

@returns `null` if the array is empty.

@param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.reduceRightOrNull
*/
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun ShortArray.reduceRightOrNull(operation: (Short, acc: Short) -> Short): Short? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
Returns `null` if the array is empty.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun IntArray.reduceRightOrNull(operation: (Int, acc: Int) -> Int): Int? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *

and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun LongArray.reduceRightOrNull(operation: (Long, acc: Long) -> Long): Long? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun FloatArray.reduceRightOrNull(operation: (Float, acc: Float) -> Float): Float? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun DoubleArray.reduceRightOrNull(operation: (Double, acc: Double) -> Double): Double? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun BooleanArray.reduceRightOrNull(operation: (Boolean, acc: Boolean) -> Boolean): Boolean? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharArray.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* * *
accumulator value that starts with [initial] value.\n*\n* Note that `acc` value passed to [operation] function should not be mutated;\n*\n* otherwise it would affect the previous value in resulting list.\n*\n* and calculates the next accumulator value.\n*\n* @param samples.collections.Collections.Aggregates.runningFold\n
*\n*\n* Note that `acc` value passed to [operation] function should not be mutated;\n*\n* otherwise it would affect the previous value in resulting list.\n*\n* and calculates the next accumulator value.\n*\n* @param samples.collections.Collections.Aggregates.runningFold\n
*\n*\n* Note that `acc` value passed to [operation] function should not be mutated;\n*\n* otherwise it would affect the previous value in resulting list.\n*\n* and calculates the next accumulator value.
value.

* @sample samples.collections.Collections.Aggregates.runningFold

```kotlin
*\n*/
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.runningFold(initial: R,
operation: (acc: R, Float) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    var accumulator = initial\n    for (element in this) {\n        accumulator = operation(accumulator, element)\n        \n        return resultOf(accumulator)\n    }\n    return resultOf(accumulator)\n}/n

*\n*/
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.runningFold(initial: R,
operation: (acc: R, Double) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    var accumulator = initial\n    for (element in this) {\n        accumulator = operation(accumulator, element)\n        \n        return resultOf(accumulator)\n    }\n    return resultOf(accumulator)\n}/n

*\n*/
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.runningFold(initial: R,
operation: (acc: R, Boolean) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    var accumulator = initial\n    for (element in this) {\n        accumulator = operation(accumulator, element)\n        \n        return resultOf(accumulator)\n    }\n    return resultOf(accumulator)\n}/n

*\n*/
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.runningFoldIndexed(initial: R,
operation: (index: Int, acc: R, T) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        \n        return resultOf(accumulator)\n    }\n    return resultOf(accumulator)\n}/n

*\n*/
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly
public inline fun <T, R> Array<out T>.runningFoldIndexed(initial: R,
operation: (index: Int, acc: R, T) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        \n        return resultOf(accumulator)\n    }\n    return resultOf(accumulator)\n}/n
```

Open Source Used In DNAC Platform Arica 9525
ByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R>

val result = ArrayList<R>(size + 1).apply { add(initial) } var accumulator = initial
for (index in indices) { accumulator = operation(index, accumulator, this[index]) } result.add(accumulator) return result

Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value. 

Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. 

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold

ShortArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R>

val result = ArrayList<R>(size + 1).apply { add(initial) } var accumulator = initial
for (index in indices) { accumulator = operation(index, accumulator, this[index]) } result.add(accumulator) return result

Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value. 

Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. 

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold


val result = ArrayList<R>(size + 1).apply { add(initial) } var accumulator = initial
for (index in indices) { accumulator = operation(index, accumulator, this[index]) } result.add(accumulator) return result

Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value. 

Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. 

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold


val result = ArrayList<R>(size + 1).apply { add(initial) } var accumulator = initial
for (index in indices) { accumulator = operation(index, accumulator, this[index]) } result.add(accumulator) return result

Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value. 

Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. 

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold


val result = ArrayList<R>(size + 1).apply { add(initial) } var accumulator = initial
for (index in indices) { accumulator = operation(index, accumulator, this[index]) } result.add(accumulator) return result

Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original array and current accumulator value that starts with [initial] value. 

Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. 

@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@sample samples.collections.Collections.Aggregates.runningFold
next accumulator value.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun <R>
        DoubleArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: Double) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) { // accumulator = operation(index, accumulator, this[index])
        accumulator = operation(accumulator, acc)
    }
    return result
}
```

The above code snippet demonstrates how to use the `runningFoldIndexed` function for a `DoubleArray`. It takes an initial value and an operation function as parameters. The operation function is applied to each element of the array, along with its index and the accumulator value from the previous step. The result is a list containing the successive accumulation values.

This function is part of the Kotlin standard library and is useful for processing arrays in a functional way. It's particularly handy when you need to perform calculations that depend on the previous element's value for the current one.
Open Source Used In DNAC Platform Arica 9528

```kotlin
// List<Short>
val result = ArrayList<Short>(size).apply { add(accumulator) }
for (index in 1 until size) {
    accumulator = operation(accumulator, this[index])
    result.add(accumulator)
}
return result
```

```kotlin
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun IntArray.runningReduce(operation: (acc: Int, Int) -> Int): List<Int> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun LongArray.runningReduce(operation: (acc: Long, Long) -> Long): List<Long> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Long>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun FloatArray.runningReduce(operation: (acc: Float, Float) -> Float): List<Float> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Float>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.runningReduce(operation: (acc: Double, Double) -> Double): List<Double> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Double>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.runningReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Boolean>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this array.
 *<param>[operation]</param>
 *function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun CharArray.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Open Source Used In DNAC Platform Arica 9528
Open Source Used In DNAC Platform Arica  9529

(acc: Char, Char) -> Char): List<Char> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Char>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n} \n\n**Returns a list containing successive accumulation values generated by applying \[operation\] from left to right on each element, its index in the original array and current accumulator value that starts with the first element of this array.**

Note that `acc` value passed to \[operation\] function should not be mutated; otherwise it would affect the previous value in resulting list.

\[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

**@param \[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.**

**@sample samples.collections.Collections.Aggregates.runningReduce\n**

*\n@SinceKotlin("1.4")@kotlin.internal.InlineOnly

public inline fun ByteArray.runningReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Byte>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n} \n
**Returns a list containing successive accumulation values generated by applying \[operation\] from left to right on each element, its index in the original array and current accumulator value that starts with the first element of this array.**

\[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

**@param \[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.**

**@sample samples.collections.Collections.Aggregates.runningReduce\n**

*\n@SinceKotlin("1.4")@kotlin.internal.InlineOnly

public inline fun ShortArray.runningReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Short>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n} \n
**Returns a list containing successive accumulation values generated by applying \[operation\] from left to right on each element, its index in the original array and current accumulator value that starts with the first element of this array.**

\[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

**@param \[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.**

**@sample samples.collections.Collections.Aggregates.runningReduce\n**

*\n@SinceKotlin("1.4")@kotlin.internal.InlineOnly

public inline fun IntArray.runningReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Int>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n} \n
**Returns a list containing successive accumulation values generated by applying \[operation\] from left to right on each element, its index in the original array and current accumulator value that starts with the first element of this array.**

\[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

**@param \[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.**

**@sample samples.collections.Collections.Aggregates.runningReduce\n**

*\n@SinceKotlin("1.4")@kotlin.internal.InlineOnly

public inline fun LongArray.runningReduceIndexed(operation: (index: Int, acc: Long, Long) -> Long): List<Long> {\n    if (isEmpty()) return emptyList()\n    var accumulator = this[0]\n    val result = ArrayList<Long>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n} \n
**Returns a list containing successive accumulation values generated by applying \[operation\] from left to right on each element, its index in the original array and current accumulator value that starts with the first element of this array.**

\[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

**@param \[operation\] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.**

**@sample samples.collections.Collections.Aggregates.runningReduce\n**
result.add(accumulator)\n  }\n  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.runningReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<Float>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.runningReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<Double>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.runningReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<Boolean>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.runningReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<Char>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun StringArray.runningReduceIndexed(operation: (index: Int, acc: String, String) -> String): List<String> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<String>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}\n/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.4")\n@kotlin.internal.InlineOnly\npublic inline fun Array<out Any>.runningReduceIndexed(operation: (index: Int, acc: Any, Any) -> Any): List<Any> {
  if (isEmpty()) return emptyList()
  var accumulator = this[0] \n  val result = ArrayList<Any>(size).apply {
    add(accumulator) }
  for (index in 1 until size) {
    accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
  }
  return result
}
resulting list.

- \* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
- \* @sample samples.collections.Collections.Aggregates.scan

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> ByteArray.scan(initial: R, operation: (acc: R, Byte) -> R): List<R> { \n  return runningFold(initial, operation)\n}

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> ShortArray.scan(initial: R, operation: (acc: R, Short) -> R): List<R> { \n  return runningFold(initial, operation)\n}

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> IntArray.scan(initial: R, operation: (acc: R, Int) -> R): List<R> { \n  return runningFold(initial, operation)\n}

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> LongArray.scan(initial: R, operation: (acc: R, Long) -> R): List<R> { \n  return runningFold(initial, operation)\n}

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> FloatArray.scan(initial: R, operation: (acc: R, Float) -> R): List<R> { \n  return runningFold(initial, operation)\n}

- \*\* @Since Kotlin("1.4\")\*\* @WasExperimental(ExperimentalStdlibApi::class)\*\* @kotlin.internal.InlineOnly\*\* public inline fun <R> DoubleArray.scan(initial: R, operation: (acc: R, Double) -> R): List<R> { \n  return runningFold(initial, operation)\n}

Open Source Used In DNAC Platform Arica 9531
public inline fun <R> BooleanArray.scan(initial: R, operation: (acc: R, Boolean) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
public inline fun <T, R> Array<out T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> ByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
    return runningFoldIndexed(initial, operation)
}

/*
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <R> ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

/*
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

/*
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

/*
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}

*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
    return runningFoldIndexed(initial, operation)
}
    return runningFoldIndexed(initial, operation)\n\n    * Note that `acc` value passed to [operation] function should not be mutated;\n    * otherwise it would affect the previous value in resulting list.\n\n    * @param [operation] function that takes the index of an element, current accumulator value of any type and the element itself, and calculates the next accumulator value.\n\n    * @sample samples.collections.Collections.Aggregates.scan

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <R> FloatArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n    * Note that `acc` value passed to [operation] function should not be mutated;\n    * otherwise it would affect the previous value in resulting list.\n\n    * @param [operation] function that takes the index of an element, current accumulator value of any type and the element itself, and calculates the next accumulator value.\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <R> DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n    * Note that `acc` value passed to [operation] function should not be mutated;\n    * otherwise it would affect the previous value in resulting list.\n\n    * @param [operation] function that takes the index of an element, current accumulator value of any type and the element itself, and calculates the next accumulator value.\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <R> BooleanArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n    * Note that `acc` value passed to [operation] function should not be mutated;\n    * otherwise it would affect the previous value in resulting list.\n\n    * @param [operation] function that takes the index of an element, current accumulator value of any type and the element itself, and calculates the next accumulator value.\n
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun <R> CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    return runningFoldIndexed(initial, operation)\n\n    * Note that `acc` value passed to [operation] function should not be mutated;\n    * otherwise it would affect the previous value in resulting list.\n\n    * @param [operation] function that takes the index of an element, current accumulator value of any type and the element itself, and calculates the next accumulator value.\n
@Deprecated("Use runningReduce instead.",
            ReplaceWith("runningReduce(operation)"), level =
            DeprecationLevel.ERROR)\n    fun <S, T : S> Array<out T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
        List<@Sample(Samples.Collections.Aggregates.scan)
            *<S>\n    }

@Deprecated("Use runningReduce instead.",
            ReplaceWith("runningReduce(operation)"), level =
            DeprecationLevel.ERROR)\n    fun <S, T : S> ByteArray.scanReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
        List<@Sample(Samples.Collections.Aggregates.scan)
            *<S>\n    }

@Deprecated("Use runningReduce instead.",
            ReplaceWith("runningReduce(operation)"), level =
            DeprecationLevel.ERROR)\n    fun <S, T : S> ShortArray.scanReduce(operation: (acc: Short, Short) -> Short): List<Short> {
        List<@Sample(Samples.Collections.Aggregates.scan)
            *<S>\n    }
DeprecationLevel.ERROR)\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.scanReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double>
    return runningReduceIndexed(operation)\n\nDeprecationLevel.ERROR)\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.scanReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean>
    return runningReduceIndexed(operation)\n\nDeprecationLevel.ERROR)\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char>
    return runningReduceIndexed(operation)\n
/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun <T> Array<out T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun <T> Array<out T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumByDouble(selector: (Short) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumByDouble(selector: (Float) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumByDouble(selector: (Double) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
FloatArray.sumByDouble(selector: (Float) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumByDouble(selector: (Double) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumOf(selector: (Byte) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumOf(selector: (Short) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumOf(selector: (Int) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumOf(selector: (Long) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumOf(selector: (Float) -> Double): Double {
    var sum = 0.0
    for (element in this)
        sum += selector(element)
    return sum
}
BooleanArray.sumOf(selector: (Boolean) -> Double): Double {
    var sum = Double(0.0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

CharArray.sumOf(selector: (Char) -> Double): Double {
    var sum = Double(0.0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

Array<out T>.sumOf(selector: (T) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

ByteArray.sumOf(selector: (Byte) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

ShortArray.sumOf(selector: (Short) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

IntArray.sumOf(selector: (Int) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

LongArray.sumOf(selector: (Long) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

FloatArray.sumOf(selector: (Float) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

DoubleArray.sumOf(selector: (Double) -> Int): Int {
    var sum = Int(0)
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

Open Source Used In DNAC Platform Arica 9537
An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfInt") @kotlin.internal.InlineOnly
public inline fun CharArray.sumOf(selector: (Char) -> Int): Int {
    var sum: Int = 0.toInt()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun <T> Array<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun ShortArray.sumOf(selector: (Short) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun IntArray.sumOf(selector: (Int) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun LongArray.sumOf(selector: (Long) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun FloatArray.sumOf(selector: (Float) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun DoubleArray.sumOf(selector: (Double) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun BooleanArray.sumOf(selector: (Boolean) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.

An @SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfLong") @kotlin.internal.InlineOnly
public inline fun CharArray.sumOf(selector: (Char) -> Long): Long {
    var sum: Long = 0.toLong()
    return sum
}

*/// Returns the sum of all values produced by [selector] function applied to each element in the array.
ByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <T> Array<out T>.sumOf(selector: (T) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumOf(selector: (Byte) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumOf(selector: (Short) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.sumOf(selector: (Int) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.sumOf(selector: (Long) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.sumOf(selector: (Float) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.sumOf(selector: (Double) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.sumOf(selector: (Boolean) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\nByLambdaReturnType\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.sumOf(selector: (Char) -> ULong): ULong {\n    var sum: ULong = 0.toULong() for (element in this) {\n        sum += selector(element)\n    } return sum\n} // * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.\n
@Suppress("UNCHECKED_CAST")\nreturn this as Array<T>\n} // * Splits the original array into pair of lists,\n// * where *first* list contains elements for which [predicate] yielded `true`, \n// * while *second* list
contains elements for which [predicate] yielded `false`.

```kotlin
@sample
```

```
samples.collections.Arrays.Transformations.partitionArrayOfPrimitives

@sample
```

```
// Splits the original array into pair of lists,\n* where *first* list contains elements for which [predicate] yielded `true`,\n* while *second* list contains elements for which [predicate] yielded `false`.\n
```kotlin
@sample
```

```
samples.collections.Arrays.Transformations.partitionArrayOfPrimitives

@sample
```

```
// Splits the original array into pair of lists,\n* where *first* list contains elements for which [predicate] yielded `true`,\n* while *second* list contains elements for which [predicate] yielded `false`.\n
```kotlin
@sample
```

```
samples.collections.Arrays.Transformations.partitionArrayOfPrimitives

@sample
```

```
// Splits the original array into pair of lists,\n* where *first* list contains elements for which [predicate] yielded `true`,\n* while *second* list contains elements for which [predicate] yielded `false`.\n
```kotlin
@sample
```

```
samples.collections.Arrays.Transformations.partitionArrayOfPrimitives

@sample
```

```
// Splits the original array into pair of lists,\n* where *first* list contains elements for which [predicate] yielded `true`,\n* while *second* list contains elements for which [predicate] yielded `false`.\n
```kotlin
@sample
```

```
samples.collections.Arrays.Transformations.partitionArrayOfPrimitives

@sample
```

```
// Splits the original array into pair of lists,\n* where *first* list contains elements for which [predicate] yielded `true`,\n* while *second* list contains elements for which [predicate] yielded `false`.\n
```kotlin
@sample
```
elements for which `[predicate]` yielded `true`, *while* `[second]` list contains elements for which `[predicate]` yielded `false`. *The returned list has length of the shortest collection.*

```kotlin
inline fun CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {
    val first = ArrayList<Char>()
    val second = ArrayList<Char>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
infix fun <T, R> Array<out T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> ByteArray.zip(other: Array<out R>): List<Pair<Byte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> IntArray.zip(other: Array<out R>): List<Pair<Int, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> LongArray.zip(other: Array<out R>): List<Pair<Long, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> BooleanArray.zip(other: Array<out R>): List<Pair<Boolean, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <R> CharArray.zip(other: Array<out R>): List<Pair<Char, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
infix fun <T, R, V> Array<out T>.zipIterable(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
infix fun <R> LongArray.zipIterable(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```
of  `this` array and the [other] array with the same index
* using the provided [transform] function applied to each
pair of elements.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
* The returned list has length of the shortest collection.
returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<R> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Int> IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Byte> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Char> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Long> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Float> FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Double> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Boolean> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Char> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Byte> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Short> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Int> IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Long> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Double> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Boolean> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Char> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Byte> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Short> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Long> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Double> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Boolean> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Char> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Byte> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Short> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Long> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Double> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.

* The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable<Boolean> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n    \{ t1, t2 -> t1 to t2 \}\n}\n
* Returns a list of pairs built from the elements of 'this' collection and [other] array with the same index.
if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public inline fun <R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public inline fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public inline fun <R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public infix fun ByteArray.zip(other: ByteArray): List<Pair<Byte, Byte>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public infix fun ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public infix fun IntArray.zip(other: IntArray): List<Pair<Int, Int>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Operations.zipIterableWithTransform
 */
public infix fun LongArray.zip(other: LongArray): List<Pair<Long, Long>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

Open Source Used In DNAC Platform Arica 9545
returned list has length of the shortest collection.

samples.collections.Iterables.Operations.zipIterable

* @sample 


List<Pair<Float, Float>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} 

* Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.

The returned list has length of the shortest collection.

samples.collections.Iterables.Operations.zipIterable fun DoubleArray.zip(other: DoubleArray): 

List<Pair<Double, Double>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} 

* Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.

The returned list has length of the shortest collection.

samples.collections.Iterables.Operations.zipIterable fun BooleanArray.zip(other: BooleanArray): 

List<Pair<Boolean, Boolean>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} 

* Returns a list of values built from the elements of `this` array and the [other] array with the same index.

The returned list has length of the shortest collection.

samples.collections.Iterables.Operations.zipIterable fun CharArray.zip(other: CharArray): 

List<Pair<Char, Char>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} 

* Returns a list of values built from the elements of `this` array and the [other] array with the same index.

The returned list has length of the shortest collection.

samples.collections.Iterables.Operations.zipIterable fun ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): 

List<V> { 
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
} 

* Returns a list of values built from the elements of `this` array and the [other] array using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest array.

samples.collections.Iterables.Operations.zipIterableWithTransform fun <V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): 

List<V> { 
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
} 

* Returns a list of values built from the elements of `this` array and the [other] array using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest array.


List<Int> { 
    val size = minOf(size, other.size)
    val list = ArrayList<Int>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
} 

* Returns a list of values built from the elements of `this` array and the [other] array using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest array.


List<Float> { 
    val size = minOf(size, other.size)
    val list = ArrayList<Float>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
} 

* Returns a list of values built from the elements of `this` array and the [other] array using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest array.

samples.collections.Iterables.Operations.zipIterableWithTransform fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): 

List<Double> { 
    val size = minOf(size, other.size)
    val list = ArrayList<Double>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
} 

* Returns a list of values built from the elements of `this` array and the [other] array using the provided [transform] function applied to each pair of elements.

The returned list has length of the shortest array.
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * @sample

samples.collections.Iterables.Operations.zipIterableWithTransform\n *\n\npublic inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n\npublic inline fun <V> List<V>.zipIterableWithTransform\n *\n\npublic inline fun <V> String.zipIterableWithTransform\n *\n\npublic inline fun <V> CharArray.zipIterableWithTransform\n *\n\npublic inline fun <V> IntArray.zipIterableWithTransform\n *\n
```kotlin
public inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
public inline fun <V> List<V>.zipIterableWithTransform\n *\n\npublic inline fun <V> String.zipIterableWithTransform\n *\n\npublic inline fun <V> CharArray.zipIterableWithTransform\n *\n\npublic inline fun <V> IntArray.zipIterableWithTransform\n *
```
buffer.append(transform(element))
else
    buffer.append(element.toString())
} else break


buffer.append(truncated)
buffer.append(postfix)
return

buffer()n*/n* * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. n * n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]n * elements will be appended, followed by the [truncated] string (which
defaults to "..."), n * n * @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): A {

    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            break
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(truncated)
    buffer.append(postfix)
    return

buffer()n*/n* * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. n * n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]n * elements will be appended, followed by the [truncated] string (which
defaults to "..."), n * n * @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): A {

    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            break
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

buffer()n*/n* * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. n * n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]n * elements will be appended, followed by the [truncated] string (which
defaults to "..."), n * n * @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): A {

    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            break
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

buffer()n*/n* * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. n * n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]n * elements will be appended, followed by the [truncated] string (which
defaults to "..."), n * n * @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> BooleanArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): A {

    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            break
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

buffer()n*/n* * Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. n * n * If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit]n * elements will be appended, followed by the [truncated] string (which
defaults to "..."), n * n * @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> CharArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): A {

    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            break
    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(truncated)
    buffer.append(postfix)
    return


null): A `Buffer` can be created as follows:

```kotlin
Buffer(buffer.append(prefix) var count = 0n for (element in this) {
    if (++count > 1)
    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element))
        else
            buffer.append(element)
    } else break
} if (limit >= 0 && count > limit)
    buffer.append(truncated)
}
```

This method takes the following parameters:

- `prefix`: The prefix to append to each element.
- `separator`: A delimiter between elements.
- `postfix`: The suffix to append to the last element.
- `limit`: The maximum number of elements to append. A negative value means no limit.
- `truncated`: The string used when the limit is exceeded.
- `transform`: A transformation function applied to each element.

The method appends each element, separated by `separator`, and appends a prefix and a postfix if required. If the limit is exceeded, it appends a truncated string. If the collection is huge, it may append only the first `limit` elements, followed by a truncated string.

Here are some examples of how to use the `joinToString` method for different collection types:

1. **For a generic array:**
   ```kotlin
   fun joinToString<T>(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```

2. **For a `ByteArray`:**
   ```kotlin
   fun ByteArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Byte) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```

3. **For a `ShortArray`:**
   ```kotlin
   fun ShortArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```

4. **For an `IntArray`:**
   ```kotlin
   fun IntArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```

5. **For a `LongArray`:**
   ```kotlin
   fun LongArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```

6. **For a `FloatArray`:**
   ```kotlin
   fun FloatArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): String {
       return joinToString(separator, prefix, postfix, limit, truncated, transform).toString()
   }
   ```
append, followed by the [truncated] string (which defaults to "...").

```java
public fun DoubleArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}
```

```java
/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first
 * [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun BooleanArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = ",", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}
```

```java
/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first
 * [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun CharArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}
```

```java
/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Array<out T>.asIterable(): Iterable<T> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}
```

```java
/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections-Sequences.Building.sequenceFromArray
 */
public fun <T> Array<out T>.asSequence(): Sequence<T> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.

```kotlin
public fun IntArray.asSequence(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun FloatArray.asSequence(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun DoubleArray.asSequence(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun BooleanArray.asSequence(): Sequence<Boolean> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun CharArray.asSequence(): Sequence<Char> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfByte")
public fun Array<out Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfShort")
public fun Array<out Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfInt")
public fun Array<out Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfLong")
public fun Array<out Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfFloat")
public fun Array<out Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
@kotlin.jvm.JvmName("averageOfDouble")
public fun Array<out Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun ByteArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun ShortArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun IntArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun LongArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun FloatArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun DoubleArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
public fun BooleanArray.asSequence(): Sequence<Boolean> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
public fun CharArray.asSequence(): Sequence<Char> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
@sample samples.collections.Sequences.Building.sequenceFromArray
```
```kotlin
/**
 * Returns an average value of elements in the array.
 */
public fun DoubleArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Array<out Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Array<out Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Array<out Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Array<out Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Array<out Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Array<out Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    try {
        return random.nextInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun LongRange.random(): Long {
    try {
        return random.nextLong(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun CharRange.random(): Char {
    return random(CharRange.random())
}

/**
 * Returns a random element from the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    try {
        return random.nextInt(this)
    } catch(e: RandomnessException) {
        throw IllegalStateException(e.message)
    }
}
```

---

Open Source Used In DNAC Platform Arica 9552
removed.

@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Double): Boolean {
    return value.toIntExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Double): Boolean {
    return value.toLongExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Double): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Double): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Float): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Float): Boolean {
    return value.toLongExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Float): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Float): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Byte): Boolean {
    return value.toIntExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Byte): Boolean {
    return value.toLongExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Byte): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Byte): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@DeprecatedSinceKotlin(warningSince = "1.3", errorSince = "1.4")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Byte): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun
ClosedRange<Long>.contains(value: Int): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Int): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Int): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Int): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Int): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Long): Boolean {
    return value.toIntExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Long): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Long): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Long): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let {
        if (it != null) contains(it) else false
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Short): Boolean {
Short): Boolean {
    return contains(value.toFloat())
} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Short): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Char): Boolean {
    return contains(value.toFloat())
} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Short): Boolean {
    return contains(value.toFloat())
} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Int): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Char): CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Char): CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Char): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */

Char): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)\n} /*
   * Returns a progression from this value down to the specified [to] value with the step -1.0 * \n * The [to] value should be less than or equal to ‘this’ value.\n * If the [to] value is greater than ‘this’ value the returned progression is empty.\n */
progression from this value down to the specified [to] value with the step -1.

* The [to] value should be less than or equal to `this` value.

* If the [to] value is greater than `this` value the returned progression is empty.

*/

public infix fun Long.downTo(to: Short): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */

public infix fun Byte.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */

public infix fun Short.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.
 */

public fun IntProgression.reversed(): IntProgression {
    return IntProgression.fromClosedRange(last, first, -step)
}

/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.
 */

public fun LongProgression.reversed(): LongProgression {
    return LongProgression.fromClosedRange(last, first, -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */

public infix fun IntProgression.step(step: Int): IntProgression {
    checkStepIsPositive(step > 0, step)
    return IntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */

public infix fun LongProgression.step(step: Long): LongProgression {
    checkStepIsPositive(step > 0, step)
    return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */

public infix fun CharProgression.step(step: Int): CharProgression {
    checkStepIsPositive(step > 0, step)
    return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

internal fun Int.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null
}

internal fun Long.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null
}

internal fun Short.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null
}

internal fun Double.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toInt().toByte() else null
}

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toInt().toByte() else null
}

internal fun Long.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun Float.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null
}

internal fun Long.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toLong()..Short.MAX_VALUE.toLong()) this.toShort() else null
}

internal fun Double.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toDouble()..Short.MAX_VALUE.toDouble()) this.toShort() else null
}

internal fun Float.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toFloat()..Short.MAX_VALUE.toFloat()) this.toShort() else null
}
value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.

```kotlin
@sample samples.comparisons.ComparableOps.coerceAtLeast
public fun <T : Comparable<T>>
T.coerceAtLeast(minimumValue: T): T {
    return if (this < minimumValue) minimumValue else this
} /* Ensures that this value is not less than the specified [minimumValue]. */

@sample samples.comparisons.ComparableOps.coerceAtMost
public fun <T : Comparable<T>>
T.coerceAtMost(maximumValue: T): T {
    return if (this > maximumValue) maximumValue else this
} /* Ensures that this value is not greater than the specified [maximumValue]. */
```

### Byte

```kotlin
Byte.coerceAtLeast(minimumValue: Byte): Byte {
    return if (this < minimumValue) minimumValue else this
}

Byte.coerceAtMost(maximumValue: Byte): Byte {
    return if (this > maximumValue) maximumValue else this
}
```

### Short

```kotlin
Short.coerceAtLeast(minimumValue: Short): Short {
    return if (this < minimumValue) minimumValue else this
}

Short.coerceAtMost(maximumValue: Short): Short {
    return if (this > maximumValue) maximumValue else this
}
```

### Int

```kotlin
Int.coerceAtLeast(minimumValue: Int): Int {
    return if (this < minimumValue) minimumValue else this
}

Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}
```

### Long

```kotlin
Long.coerceAtLeast(minimumValue: Long): Long {
    return if (this < minimumValue) minimumValue else this
}

Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}
```

### Float

```kotlin
Float.coerceAtLeast(minimumValue: Float): Float {
    return if (this < minimumValue) minimumValue else this
}

Float.coerceAtMost(maximumValue: Float): Float {
    return if (this > maximumValue) maximumValue else this
}
```

### Double

```kotlin
Double.coerceAtLeast(minimumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}

Double.coerceAtMost(maximumValue: Double): Double {
    return if (this > maximumValue) maximumValue else this
}
```
Ensures that this value lies in the specified range \[minimumValue..maximumValue\].

* Ensures that this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's less than or equal to the \[maximumValue\] or the \[maximumValue\] otherwise.
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's less than \[minimumValue\], or \[maximumValue\] if this value is greater than \[maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].
  * 
  * @return this value if it's in the range, or \[minimumValue\] if this value lies in the specified range \[minimumValue..maximumValue\].

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

IllegalArgumentOutOfRangeException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
public fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {
    if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
minimum $minimumValue.")
    return if (this < minimumValue) return minimumValue
    else if (this > maximumValue) return maximumValue
    else return this
}

/**
 * Ensures that this value lies in the specified 
range.
 * 
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`. * * * @sample
samples.comparisons.ComparableOps.coerceInFloatingPointRange
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw
    IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        // this < start equiv to this <= start && !(this >= start)
        range.lessThanOrEquals(this, range.start) &&
        !range.lessThanOrEquals(range.start, this) -> range.start
        // this > end equiv to this >= end && !(this <= end)
        range.lessThanOrEquals(range.endInclusive, this) && !range.lessThanOrEquals(this, range.endInclusive) ->
        range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified 
range.
 * 
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`. * * * @sample
samples.comparisons.ComparableOps.coerceInComparable
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified 
range.
 * 
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`. * * * @sample
samples.comparisons.ComparableOps.coerceIn
 */
public fun Int.coerceIn(range: ClosedRange<Int>): Int {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Int>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified 
range.
 * 
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive`
if this value is greater than `range.endInclusive`. * * * @sample
samples.comparisons.ComparableOps.coerceIn
 */
public fun Long.coerceIn(range: ClosedRange<Long>): Long {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Long>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

Open Source Used In DNAC Platform Arica 9561
it's less than other,
* or a positive number if it's greater than other.\n *\n@kotlin.internal.InlineOnly
@Suppress("OVERRIDE_BY_INLINE")
public override inline operator fun compareTo(other: UByte): Int =
this.toInt().compareTo(other.toInt())\n/**\n* Compares this value with the specified value for order.\n* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n* or a positive number if it's greater than other.\n*/
@kotlin.internal.InlineOnly
public inline operator fun compareToTo(other: UByte): Int =
this.toInt().compareToTo(other.toInt())\n/**\n* Compares this value with the specified value for order.\n* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n* or a positive number if it's greater than other.\n*/
@kotlin.internal.InlineOnly
public inline operator fun compareToTo(other: UShort): Int =
this.toInt().compareToTo(other.toInt())\n/**\n* Compares this value with the specified value for order.\n* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n* or a positive number if it's greater than other.\n*/
@kotlin.internal.InlineOnly
public inline operator fun compareToTo(other: UInt): Int =
this.toUInt().compareToTo(other)\n/**\n* Compares this value with the specified value for order.\n* Returns zero if this value is equal to the specified other value, a negative number if it's less than other,\n* or a positive number if it's greater than other.\n*/
@kotlin.internal.InlineOnly
public inline operator fun compareToTo(other: ULong): Int =
this.toULong().compareToTo(other)\n/** Adds the other value to this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): UInt =
this.toUInt().plus(other.toUInt())\n/** Adds the other value to this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun plus(other: UShort): UInt =
this.toUInt().plus(other.toUInt())\n/** Adds the other value to this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun plus(other: UInt): UInt =
this.toUInt().plus(other)\n/** Adds the other value to this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun plus(other: ULong): ULong =
this.toULong().plus(other)\n/** Subtracts the other value from this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun minus(other: UByte): UInt =
this.toUInt().minus(other.toUInt())\n/** Subtracts the other value from this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun minus(other: UShort): UInt =
this.toUInt().minus(other.toUInt())\n/** Subtracts the other value from this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): UInt =
this.toUInt().minus(other)\n/** Subtracts the other value from this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong =
this.toULong().minus(other)\n/** Multiplies this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): UInt =
this.toUInt().times(other.toUInt())\n/** Multiplies this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): UInt =
this.toUInt().times(other.toUInt())\n/** Multiplies this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): UInt =
this.toUInt().times(other)\n/** Multiplies this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong =
this.toULong().times(other)\n/** Divides this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): UInt =
this.toUInt().div(other.toUInt())\n/** Divides this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): UInt =
this.toUInt().div(other.toUInt())\n/** Divides this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): UInt =
this.toUInt().div(other)\n/** Divides this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong =
this.toULong().div(other)\n/** Calculates the remainder of dividing this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): UInt =
this.toUInt().rem(other.toUInt())\n/** Calculates the remainder of dividing this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt =
this.toUInt().rem(other.toUInt())\n/** Calculates the remainder of dividing this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt =
this.toUInt().rem(other)\n/** Calculates the remainder of dividing this value by the other value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong =
this.toULong().rem(other)\n/** Increments this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun inc(): UByte =
UByte(data.inc())\n/** Increments this value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun dec(): UByte =
UByte(data.dec())\n/** Creates a range from this value to the specified [other] value.\n*\n@kotlin.internal.InlineOnly
public inline operator fun
operator fun rangeTo(other: UByte): UIntRange = UIntRange(this.toUInt(), other.toUInt())

    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data)

    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data)

    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data)

    /** Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): UByte = UByte(data.inv())

    /** Converts this [UByte] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline fun toByte(): Byte = data

    /** Converts this [UByte] value to [Short]. */
    @kotlin.internal.InlineOnly
    public inline fun toShort(): Short = data.toShort() and 0xFF

    /** Converts this [UByte] value to [Int]. */
    @kotlin.internal.InlineOnly
    public inline fun toInt(): Int = data.toInt() and 0xFF

    /** Converts this [UByte] value to [Long]. */
    @kotlin.internal.InlineOnly
    public inline fun toLong(): Long = data.toLong() and 0xFF

    /** Converts this [UByte] value to [Float]. */
    @kotlin.internal.InlineOnly
    public inline fun toFloat(): Float = this.toInt().toFloat()

    /** Converts this [UByte] value to [Double]. */
    @kotlin.internal.InlineOnly
    public inline fun toDouble(): Double = this.toInt().toDouble()
Since Kotlin("1.3")@ExperimentalUnsignedTypes
public inline fun Short.toUByte(): UByte = UByte(this.toByte())
/**
 * Converts this [Int] value to [UByte].
 * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `Int`. The resulting `UByte` value is represented by the least significant 8 bits of this `Int` value.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline fun Int.toUByte(): UByte = UByte(this.toByte())
/**
 * Converts this [Long] value to [UByte].
 * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `Long`. The resulting `UByte` value is represented by the least significant 8 bits of this `Long` value.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline fun Long.toUByte(): UByte = UByte(this.toByte())

// Auto-generated file. DO NOT EDIT!

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline class UInt @PublishedApi internal constructor(@PublishedApi internal val data: Int) : Comparable<UInt> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UInt can have.
         */
        public const val MIN_VALUE: UInt = UInt(0)

        /**
         * A constant holding the maximum value an instance of UInt can have.
         */
        public const val MAX_VALUE: UInt = UInt(-1)

        /**
         * The number of bytes used to represent an instance of UInt in a binary form.
         */
        public const val SIZE_BYTES: Int = 4

        /**
         * The number of bits used to represent an instance of UInt in a binary form.
         */
        public const val SIZE_BITS: Int = 32
    }

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public override inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())
    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt())
    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UInt): UInt = UInt(this.data.plus(other.data))
    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)

    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): UInt = this.minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): UInt = this.minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data))
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): UInt = this.times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): UInt = this.times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): UInt = UInt(this.data.times(other.data))
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = this.toULong().times(other)

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun divide(other: UByte): UInt = this.divide(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun divide(other: UShort): UInt = this.divide(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun divide(other: UInt): UInt = UInt(this.data.divide(other.data))
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun divide(other: ULong): ULong = this.toULong().divide(other)

    /** Rounds this value to the nearest value. */
    @kotlin.internal.InlineOnly
    public inline operator fun round(): UInt = this.round()
value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
* the same
numerical value as this `UInt`.*
* The resulting `UByte` value is represented by the least significant 8 bits of
this `UInt`.*
* If this value is less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents
* the same numerical value as this `UInt`.*
* The resulting `UShort` value is represented by the least significant 16 bits of this `UInt`.*
* The result of dividing this `UInt` value by 2 is an integer value.
  * The result of this division is the greatest `UShort` value that is less than or equal to
the result of `intValue / 2`.
* If this value is less than or equals to [ULong.MAX_VALUE], the resulting `ULong` value represents
* the same numerical value as this `UInt`.*
* The least significant 32 bits of this `ULong` value are the same as the bits of this `UInt`.*
* whereas the most significant 32 bits are filled with zeros.*
* The resulting `ULong` value is represented by the least significant 32 bits of this `UInt` value.*
* The least significant 8 bits of this `UInt` value are the same as the bits of this `Byte` value.*
* whereas the most significant 24 bits are filled with the sign bit of this value.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Byte.toUInt(): UInt = UInt(this.toInt())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Short.toUInt(): UInt = UInt(this.toInt())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Int.toUInt(): UInt = UInt(this)
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Long.toUInt(): UInt = UInt(this.toLong())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Double.toUInt(): UInt = doubleToUInt(this)
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun public inline fun toUByte(): UByte =
data.toUByte() /* Converts this [UInt] value to [UShort]. */
* The result of this division is the greatest `UShort` value that is less than or equal to
the result of `intValue / 2`.
* If this value is less than or equals to [ULong.MAX_VALUE], the resulting `ULong` value represents
* the same numerical value as this `UInt`.*
* The least significant 32 bits of this `ULong` value are the same as the bits of this `UInt`.*
* whereas the most significant 32 bits are filled with zeros.*
* The resulting `ULong` value is represented by the least significant 32 bits of this `UInt` value.*
* The least significant 8 bits of this `UInt` value are the same as the bits of this `Byte` value.*
* whereas the most significant 24 bits are filled with the sign bit of this value.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Byte.toUInt(): UInt = UInt(this.toInt())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Short.toUInt(): UInt = UInt(this.toInt())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Int.toUInt(): UInt = UInt(this)
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Long.toUInt(): UInt = UInt(this.toLong())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Double.toUInt(): UInt = doubleToUInt(this)
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun
constructor(@PublishedApi internal val data: Short) : Comparable<UShort> {

    companion object {
        /**
        * A constant holding the minimum value an instance of UShort can have.
        */
        public const val MIN_VALUE: UShort = UShort(0)

        /**
        * A constant holding the maximum value an instance of UShort can have.
        */
        public const val MAX_VALUE: UShort = UShort(-1)

        /**
        * The number of bytes used to represent an instance of UShort in a binary form.
        */
        public const val SIZE_BYTES: Int = 2

        /**
        * The number of bits used to represent an instance of UShort in a binary form.
        */
        public const val SIZE_BITS: Int = 16

        /**
        * A constant holding the specified value for order.
        */
        * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
        * or a positive number if it's greater than other.
        */
        public inline operator fun compareTo(other: UByte): Int = this.toUInt().compareTo(other.toInt())

        /**
        * A constant holding the specified value for order.
        */
        * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
        * or a positive number if it's greater than other.
        */
        public inline operator fun compareTo(other: UShort): Int = this.toUInt().compareTo(other.toUInt())

        /**
        * A constant holding the specified value for order.
        */
        * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
        * or a positive number if it's greater than other.
        */
        public inline operator fun compareTo(other: UInt): UInt = this.toUInt().compareTo(other)

        /**
        * A constant holding the specified value for order.
        */
        * Returns zero if this value is equal to the specified other value, a negative number if it's less than other,
        * or a positive number if it's greater than other.
        */
        public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)

        /**
        * Adds the other value to this value.
        */
        public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())

        /**
        * Adds the other value to this value.
        */
        public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())

        /**
        * Adds the other value to this value.
        */
        public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other)

        /**
        * Adds the other value to this value.
        */
        public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)

        /**
        * Subtracts the other value from this value.
        */
        public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())

        /**
        * Subtracts the other value from this value.
        */
        public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt())

        /**
        * Subtracts the other value from this value.
        */
        public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)

        /**
        * Subtracts the other value from this value.
        */
        public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)

        /**
        * Multiplies this value by the other value.
        */
        public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())

        /**
        * Multiplies this value by the other value.
        */
        public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())

        /**
        * Multiplies this value by the other value.
        */
        public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)

        /**
        * Multiplies this value by the other value.
        */
        public inline operator fun times(other: ULong): ULong = this.toULong().times(other)

        /**
        * Divides this value by the other value.
        */
        public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())

        /**
        * Divides this value by the other value.
        */
        public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())

        /**
        * Divides this value by the other value.
        */
        public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)

        /**
        * Divides this value by the other value.
        */
        public inline operator fun div(other: ULong): ULong = this.toULong().div(other)

        /**
        * Calculates the remainder of dividing this value by the other value.
        */
        public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())

        /**
        * Calculates the remainder of dividing this value by the other value.
        */
        public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())

        /**
        * Calculates the remainder of dividing this value by the other value.
        */
        public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)

        /**
        * Calculates the remainder of dividing this value by the other value.
        */
        public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)}
value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other.toULong())
/** Increments this value. */
@kotlin.internal.InlineOnly
public inline operator fun inc(): UShort = UShort(data.inc())
/** Decrements this value. */
@kotlin.internal.InlineOnly
public inline operator fun dec(): UShort = UShort(data.dec())
/** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: UShort): UShort = UShort(this.data and other.data)
/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: UShort): UShort = UShort(this.data or other.data)
/** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: UShort): UShort = UShort(this.data xor other.data)
/** Creates a range from this value to the specified [other] value. */
@kotlin.internal.InlineOnly
public inline infix fun rangeTo(other: UShort): UIntRange = UIntRange(this.toUInt(), other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): Byte = data.toByte()
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: Long): Long = this.toLong().rem(other.toLong())
/** Returns this value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): Byte = data.toByte()
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Byte.toUShort(): UShort = UShort(this.toShort())
/**
 * Converts this 
 value to 

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Short.toUShort(): UShort = UShort(this)
/**
 * Converts this 
 value to 

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Int.toUShort(): UShort = UShort(this.toInt())
/**
 * Converts this 
 value to 

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Long.toUShort(): UShort = UShort(this.toLong())

/**
 * A range of values of type `Char`
 */
public class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {
    override val start: Char get() = first
    override val endInclusive: Char get() = last

    override fun contains(value: Char): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean = first > last

    override fun equals(other: Any?): Boolean =
        other is CharRange && (isEmpty() && other.isEmpty() ||
        first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())

    override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type Char. */
        public val EMPTY: CharRange = CharRange(1.toChar(), 0.toChar())
    }
}

/**
 * A range of values of type `Int`
 */
public class IntRange(start: Int, endInclusive: Int) : IntProgression(start, endInclusive, 1), ClosedRange<Int> {
    override val start: Int get() = first
    override val endInclusive: Int get() = last

    override fun contains(value: Int): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean = first > last

    override fun equals(other: Any?): Boolean =
        other is IntRange && (isEmpty() && other.isEmpty() ||
        first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first + last)

    override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type Int. */
        public val EMPTY: IntRange = IntRange(1, 0)
    }
}

/**
 * A range of values of type `Long`
 */
public class LongRange(start: Long, endInclusive: Long) : LongProgression(start, endInclusive, 1), ClosedRange<Long> {
    override val start: Long get() = first
    override val endInclusive: Long get() = last

    override fun contains(value: Long): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean = first > last

    override fun equals(other: Any?): Boolean =
        other is LongRange && (isEmpty() && other.isEmpty() ||
        first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first + last)

    override fun toString(): String = "$first-$last"
kotlin.random.Random

internal object EmptyIterator : ListIterator<Nothing> {
    override fun hasNext(): Boolean = false
    override fun hasPrevious(): Boolean = false
    override fun nextIndex(): Int = 0
    override fun previousIndex(): Int = -1
    override fun next(): Nothing = throw NoSuchElementException()
    override fun previous(): Nothing = throw NoSuchElementException()
}

internal object EmptyList : List<Nothing>, Serializable, RandomAccess {
    private const val serialVersionUID: Long = -7390468764508069838L

    override fun equals(other: Any?): Boolean = other is List<*> && other.isEmpty()
    override fun hashCode(): Int = 1
    override fun toString(): String = "[]"

    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun contains(element: Nothing): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()

    override fun get(index: Int): Nothing = throw IndexOutOfBoundsException("Empty list doesn't contain element at index $index.")
    override fun indexOf(element: Nothing): Int = -1
    override fun lastIndexOf(element: Nothing): Int = -1

    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun listIterator(): ListIterator<Nothing> = EmptyIterator
    override fun listIterator(index: Int): ListIterator<Nothing> {
        if (index != 0) throw IndexOutOfBoundsException("Index: $index")
        return EmptyIterator
    }

    override fun subList(fromIndex: Int, toIndex: Int): List<Nothing> {
        if (fromIndex == 0 && toIndex == 0) return this
        throw IndexOutOfBoundsException("fromIndex: $fromIndex, toIndex: $toIndex")
    }

    private fun readResolve(): Any = EmptyList
}

internal fun <T> Array<out T>.asCollection(): Collection<T> = ArrayAsCollection(this)

private class ArrayAsCollection<T>(val values: Array<out T>, val isVarargs: Boolean) : Collection<T> {
    override val size: Int get() = values.size
    override fun isEmpty(): Boolean = values.isEmpty()
    override fun contains(element: T): Boolean = values.contains(element)
    override fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) }
    override fun iterator(): Iterator<T> = values.iterator()
    // override hidden toArray implementation to prevent copying of values array
    public fun toArray(): Array<out Any?> = values.copyToArrayOfAny(isVarargs)
}

/**
 * Returns an empty read-only list.  The returned list is serializable (JVM).
 */
@kotlin.internal.InlineOnly
public fun <T> emptyList(): List<T> = EmptyList

/**
 * Returns a new read-only list of given elements.  The returned list is serializable (JVM).
 */
@kotlin.internal.InlineOnly
public fun <T> listOf(vararg elements: T): List<T> = if (elements.size > 0) elements.asList() else emptyList()

/**
 * Returns a new read-only list of single given element, if it is not null, or empty list if the element is null.  The
 * returned list is serializable (JVM).
 */
public fun <T : Any> listOfNotNull(element: T?): List<T> = if (element != null) listOf(element) else emptyList()

/**
 * Returns a new read-only list only of those given elements, that are not null.  The returned list is serializable
 * (JVM).
 */
public fun <T : Any> listOfNotNull(vararg elements: T?): List<T> = elements.filterNotNull()
function [init] is called for each list element sequentially starting from the first one. It should return the value for a list element given its index.

* @sample samples.collections.Collections.Lists.readOnlyListFromInitializer

**A反正Kotlin("1.1")**\@kotlin.internal.InlineOnly\public inline fun <T> List(size: Int, init: (index: Int) -> T): List<T> = MutableList(size, init)

**A反正Kotlin("1.1")**\@kotlin.internal.InlineOnly\public inline fun <T> MutableList(size: Int, init: (index: Int) -> T): MutableList<T> = MutableList(size, init)

**A反正Kotlin("1.3")**\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\public inline fun <E> buildList(@BuilderInference builderAction: MutableList<@BuilderInference BuilderActionType>.() -> Unit): List<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) } \n    return buildListInternal(builderAction)\n}

**A反正Kotlin("1.3")**\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\public inline fun <E> buildList(capacity: Int, @BuilderInference builderAction: MutableList<@BuilderInference BuilderActionType>.() -> Unit): List<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) } \n    return buildListInternal(capacity, builderAction)\n}

**A反正Kotlin("1.3")**\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\public inline fun <T> Collection<@BuilderInference BuilderActionType*.() -> Unit>.indicesOfCollection(): IntRange = 0..size - 1

**A反正Kotlin("1.3")**\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\public inline fun <T> Collection<@BuilderInference BuilderActionType*.() -> Unit>.lastIndex: Int

**A反正Kotlin("1.3")**@kotlin.internal.InlineOnly fun <@BuilderInference BuilderActionType*>.notEmpty(): Boolean = isNotEmpty()!!

**A反正Kotlin("1.3")**@kotlin.internal.InlineOnly fun <@BuilderInference BuilderActionType*.() -> Unit>.isEmpty(): Boolean = isEmpty()!!

**A反正Kotlin("1.3")**@kotlin.internal.InlineOnly fun <@BuilderInference BuilderActionType*.() -> Unit>.notNull(): Boolean = isNull()!!

**A反正Kotlin("1.3")**@kotlin.internal.InlineOnly fun <@BuilderInference BuilderActionType*.() -> Unit>.orElse(): Collection<T> = this ?: emptyList()
elements in the specified collection are contained in this collection.

* * * Allows to overcome type-safety restriction of `containsAll` that requires to pass a collection of type `Collection<E>`.

* @sample

samples.collections.Collections.Collections.collectionContainsAll

*@n@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases

*kotlin.internal.InlineOnly

@public inline fun <T> Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)

* * * Returns a new list with the elements of this list randomly shuffled using the specified `[random]` instance as the source of randomness.

* * @sample

samples.collections.Collections.Collections.collectionContainsAll

*/

@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases

@kotlin.internal.InlineOnly

public inline fun <@kotlin.internal.OnlyInputTypes T>

Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)

/**
 * Returns a new list with the elements of this list randomly shuffled using the specified `[random]` instance as the source of randomness.
 * The list is expected to be sorted into ascending order according to the Comparable natural ordering of its elements.
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified `[element]`, there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element, if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.
 */

public fun <T> Iterable<T>.shuffled(random: Random): List<T> =

toMutableList().apply {
    shuffle(random)
} List<T>.optimizeReadOnlyList() = when (size) {

    0 -> emptyList()
    1 -> listOf(this[0])
    else -> this
}

/**
 * Searches this list or its range for the provided `[element]` using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the Comparable natural ordering of its elements.
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified `[element]`, there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element, if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.
 */

public fun <T : Comparable<T>>

List<T?>.binarySearch(element: T?, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val cmp = compareValues(mid, element)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}

/**
 * Searches this list or its range for the provided `[element]` using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the specified `[comparator]`, otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified `[element]`, there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element, if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted according to the specified `[comparator]`.
 */

public fun <T> List<T>.binarySearch(element: T, comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val cmp = comparator.compare(mid, element)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}

/**
 * Searches this list or its range for an element having the key returned by the specified `[selector]` function equal to the provided `[key]` value using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements.
 * otherwise the result is undefined.
 * If the list contains multiple elements with the specified [key], there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element with the specified [key], if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.
 */

public inline fun <T, K : Comparable<K>>

List<T>.binarySearchBy(key: K?, fromIndex: Int = 0, toIndex: Int = size, crossinline selector: (T) -> K?) =

binarySearch(fromIndex, toIndex) { compareValues(selector(it), key) }

// do not introduce this
overload --- too rare

```kotlin
public fun <T, K> List<T>.binarySearchBy(key: K, comparator: Comparator<K>,
    fromIndex: Int = 0, toIndex: Int = size, selector: (T) -> K): Int =
    binarySearch(fromIndex, toIndex)

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkIndexOverflow(index: Int):
    Int

@PublishedApi
@SinceKotlin("1.3")
internal expect fun checkCountOverflow(count: Int):
    Int

@PublishedApi
@SinceKotlin("1.3")
internal fun throwIndexOverflow() { throw
    throwCountOverflow() { throw
    EmptyMap : Map<Any?, Nothing>, Serializable {
    override fun equals(other: Any?): Boolean = other is Map<*, *>
    override fun hashCode(): Int = 0
    override fun toString(): String = "{}"
    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun containsKey(key: Any?): Boolean = false
    override fun containsValue(value: Nothing): Boolean = false
    override fun entries(): Set<Map.Entry<Any?, Nothing>>
    override fun values(): Collection<Nothing> get() = EmptyList
    override fun keySet(): Set<Any?> = EmptySet
    override fun get(key: Any?): Nothing? = null
    fun mapOf(vararg pairs: Pair<K, V>): Map<K, V> { return
        returnCountOverflow() { throw
    
    }
pairs.toMap(LinkedHashMap(mapCapacity(pairs.size))) else emptyMap() /* \n * Returns an empty read-only map.\n */
The returned map is serializable (JVM). /* \n * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap\n */
@kotlin.internal.InlineOnly
public inline fun <K, V> mapOf(): Map<K, V> = emptyMap() /* \n * Returns an empty new [MutableMap].\n */
The returned map preserves the entry iteration order. /* \n * @sample samples.collections.Maps.Instantiation.emptyMutableMap\n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <K, V> mutableMapOf():
    MutableMap<K, V> = LinkedHashMap() /* \n * Returns a new [MutableMap] with the specified contents, given as a list of pairs\n * where the first component is the key and the second is the value.\n */
If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs. /* \n */
Entries of the map are iterated in the order they were specified. /* \n * @sample samples.collections.Maps.Instantiation.mutableMapFromPairs\n */
public fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) } /* \n */
Returns an empty new [HashMap]. /* \n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <K, V> hashMapOf(): HashMap<K, V> = HashMap<K, V>() /* \n */
@SinceKotlin("1.1")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> =
    HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) } /* \n */
Returns an empty new [LinkedHashMap] with the specified contents, given as a list of pairs\n * where the first component is the key and the second is the value.\n */
If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs. /* \n */
Entries of the map are iterated in the order they were specified. /* \n */
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) } /* \n */
Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction] and returning a read-only map with the same key-value pairs. /* \n */
The map passed as a receiver to the [builderAction] is valid only inside that function. /* \n */
Using it outside of the function produces an unspecified behavior. /* \n */
Entries of the map are iterated in the order they were added by the [builderAction]. /* \n */
/* \n * @sample samples.collections.Builders.Maps.buildMapSample\n */
public inline fun <K, V> buildMap(@BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return buildMapInternal(builderAction) /* \n */
Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction] and returning a read-only map with the same key-value pairs. /* \n */
The map passed as a receiver to the [builderAction] is valid only inside that function. /* \n */
Using it outside of the function produces an unspecified behavior. /* \n */
Entries of the map are iterated in the order they were added by the [builderAction]. /* \n */
@throws IllegalArgumentException if the given [capacity] is negative. /* \n */
@sample samples.collections.Builders.Maps.buildMapSample\n} /* \n */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <K, V> buildMap(@BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return buildMapInternal(builderAction) /* \n */
Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction] and returning a read-only map with the same key-value pairs. /* \n */
The map passed as a receiver to the [builderAction] is valid only inside that function. /* \n */
Using it outside of the function produces an unspecified behavior. /* \n */
Entries of the map are iterated in the order they were added by the [builderAction]. /* \n */
@throws IllegalArgumentException if the given [capacity] is negative. /* \n */
@sample samples.collections.Builders.Maps.buildMapSample\n} /* \n */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <K, V> buildMap(capacity: Int, @BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }
    return buildMapInternal(capacity, builderAction) /* \n */
Builds a new read-only [Map] by populating a [MutableMap] using the given [builderAction] and returning a read-only map with the same key-value pairs. /* \n */
The map passed as a receiver to the [builderAction] is valid only inside that function. /* \n */
Using it outside of the function produces an unspecified behavior. /* \n */
Entries of the map are iterated in the order they were added by the [builderAction]. /* \n */
@throws IllegalArgumentException if the given [capacity] is negative. /* \n */
@sample samples.collections.Builders.Maps.buildMapSample\n} /* \n */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>.getOrElse(key: K, defaultValue: () -> V): V = get(key) ?: defaultValue()

internal inline fun <K, V> Map<K, V>.getOrElseNullable(key: K, defaultValue: () -> V): V {
    val value = get(key)
    if (value == null && !containsKey(key)) {
        return defaultValue()
    } else {
        @Suppress("UNCHECKED_CAST")
        return value as V
    }
}

/**
 * Returns the value
 * for the given 
 * key or throws an exception if there is no such key in the map.
 * @throws
 * NoSuchElementException when the key is not found in the map.
 */
@SinceKotlin("1.1")
public fun <K, V> Map<K, V>.getValue(key: K): V = getOrImplicitDefault(key)

/**
 * Returns the value for the given key. If the key is not found in the map, calls
 * the [defaultValue] function, if it puts its result into the map under the given
 * key and returns it.
 * @throws
 * NoSuchElementException when the map doesn't contain a value for the specified
 * key and no explicit default value was provided for that map.
 */
    val value = get(key)
    return if (value == null) {
        val answer = defaultValue()
        put(key, answer)
        answer
    } else {
        value
    }
}

/**
 * Returns an
 * [Iterator] over the entries in the 
 * [Map].
 */
@kotlin.internal.InlineOnly

/**
 * Returns a 
 * [MutableIterator] over the mutable entries in
 * [MutableMap].
 */
@kotlin.jvm.JvmName("mutableIterator")
@kotlin.internal.InlineOnly

/**
 * Populates the given 
 * [MutableMap] with entries having the keys of this map and the
 * values obtained
 * by applying the [transform] function to each entry in this [Map].
 */
    return entries.associateByTo(destination, { it.key }, transform)
}

/**
 * Populates the given
 * [MutableMap] with entries having the keys obtained
 * by applying the [transform] function to each entry in this [Map] and the values of this map.
 * If any two entries are mapped to the equal keys, the value of the latter one will overwrite
 * the value associated with the former one.
 */
public fun <K, V, R> MutableMap<in K, in V>.putAll(pairs: Array<out Pair<K, V>>): Unit {
    for ((key, value) in pairs) {
        put(key, value)
    }
}

/**
 * Puts all the elements of the given sequence into this 
 * [MutableMap] with the first component in the pair being the key and the second the value.
 */
public fun <K, V> MutableMap<in K, in V>.putAll(pairs: Sequence<Pair<K, V>>): Unit {
    for ((key, value) in pairs) {
        put(key, value)
    }
}

/**
 * Returns a map containing all key-value pairs with keys matching the given
 * [predicate].
 */
public fun <K> Map<out K, V>.filterMap(predicate: (K) -> Boolean): Map<K, V> = entries.filter { it.key }
preserves the entry iteration order of the original map.

```kotlin
/* * @sample samples.collections.Maps.Filtering.filterKeys
 * @public inline fun <K, V> Map<out K, V>.filterKeys(predicate: (K) -> Boolean): Map<K, V> {
 * val result = LinkedHashMap<K, V>()
 * for (entry in this) {
 * if (predicate(entry.key)) {
 * result.put(entry.key, entry.value)
 * }
 * }
 * return result
 * } *//* Returns a map containing all key-value pairs with values matching the given [predicate].
 * The returned map preserves the entry iteration order of the original map.
 * */
@sample samples.collections.Maps.Filtering.filterValues
/* * @public inline fun <K, V> Map<out K, V>.filterValues(predicate: (V) -> Boolean): Map<K, V> {
 * val result = LinkedHashMap<K, V>()
 * for (entry in this) {
 * if (predicate(entry.value)) {
 * result.put(entry.key, entry.value)
 * }
 * }
 * return result
 * } *//* Returns a map containing all key-value pairs matching the given [predicate].
 * The returned map preserves the entry iteration order of the original map.
 * */
public inline fun <K, V> Map<out K, V>.filterTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {
 * for (element in this) {
 * if (predicate(element)) {
 * destination.put(element.key, element.value)
 * }
 * }
 * return destination
 * } *//* Returns a new map containing all key-value pairs not matching the given [predicate].
 * */
 * for (element in this) {
 * if (!predicate(element)) {
 * destination.put(element.key, element.value)
 * }
 * }
 * return destination
 * } *//* Returns a new map containing all key-value pairs not matching the given [predicate].
 * */
 * return filterTo(LinkedHashMap<K, V>(), predicate)
 * } *//* Appends all entries matching the given [predicate] to the mutable map given as [destination] parameter.
 * @return the destination map.
 * */
@sample samples.collections.Maps.Filtering.filterTo
/* * @public inline fun <K, V, M : MutableMap<in K, in V>> Map<out K, V>.filterTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {
 * for (element in this) {
 * if (predicate(element)) {
 * destination.put(element.key, element.value)
 * }
 * }
 * return destination
 * } *//* Returns a new map containing all key-value pairs matching the given [predicate].
 * */
 * for (element in this) {
 * if (predicate(element)) {
 * destination.put(element.key, element.value)
 * }
 * }
 * return destination
 * } *//* Returns a new map containing all key-value pairs not matching the given [predicate].
 * */
 * for (element in this) {
 * if (!predicate(element)) {
 * destination.put(element.key, element.value)
 * }
 * }
 * return destination
 * } *//* Returns a new map containing all key-value pairs not matching the given [predicate].
 * */
 * return filterTo(LinkedHashMap<K, V>(), predicate)
 * } *//* Returns a new map containing all key-value pairs matching the given [predicate].
 * */
 * return toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()
 * } *//* Returns a new read-only map containing all key-value pairs from the original collection.
 * */
private fun <K, V> Map<out K, V>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap()
```

---

Open Source Used In DNAC Platform Arica 9577
when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() }

/**
 * Returns a new mutable map containing all key-value pairs from the original map.
 * The returned map preserves the entry iteration order of the original map.
 * Creates a new read-only map by replacing or adding an entry to this map from a
given key-value pair. The returned map preserves the entry iteration order of the
original map. The [pair] is iterated in the end if it has a unique key.

@SinceKotlin("1.1")
public fun <K, V> Map<K, V>.toMutableMap(): MutableMap<K, V> = LinkedHashMap(this)

/**
 * Populates and returns the new read-only map by replacing or adding entries to this map from a
given collection of key-value pairs. The returned map preserves the entry iteration order of the original map.

@SinceKotlin("1.1")
    destination.apply {
        putAll(this@toMap) }\n
/**
 * Creates a new read-only map by replacing or adding entries to this map from a given array of key-value pairs.

@SinceKotlin("1.1")
    this.toMutableMap().apply { putAll(pairs) }

/**
 * Appends or replaces the given [pair] in this mutable map.

@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableList<in K, in V>.plusAssign(pair: Pair<K, V>) {
    put(pair.first, pair.second)
}

/**
 * Appends or replaces all pairs from the given collection of [pairs] in this mutable map.

@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableList<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Returns a map containing all entries of the original map except the given [key].

@SinceKotlin("1.1")
public fun <K, V> Map<K, V>.minus(key: K): Map<K, V> =
    this.toMutableMap().apply { minusAssign(key) }

/**
 * Returns a map containing all entries of the original map except those entries
* the keys of which are contained in the given [keys] collection.

@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }

/**
 * Optimizes the read-only map.

@kotlin.internal.InlineOnly
public operator fun <K, V>xfa/ in DNA C Platform Arica 9578
Array(out K): Map<K, V> =
    this.toMutableMap().apply { minusAssign(keys)
    }.optimizeReadOnlyMap()

/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] sequence.
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys)
    }.optimizeReadOnlyMap()

/**
 * Removes the entry with the given [key] from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public operator fun <K, V> Map(out K, V>.remove(key: K): V {
    remove(key)
}

/**
 * Removes all entries the keys of which are contained in the given [keys] collection from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public operator fun <K, V> Map(out K, V>.removeAll(keys: Iterable<K>): Map<K, V> =
    this.toMutableMap().apply { removeAll(keys)
    }.optimizeReadOnlyMap()

/**
 * Removes all entries the keys of which are contained in the given [keys] array from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public operator fun <K, V> Map(out K, V>.removeAll(keys: Array<out K>): Map<K, V> =
    this.toMutableMap().apply { removeAll(keys)
    }.optimizeReadOnlyMap()

/**
 * Removes all entries the keys of which are contained in the given [keys] sequence from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public operator fun <K, V> Map(out K, V>.removeAll(keys: Sequence<K>): Map<K, V> =
    this.toMutableMap().apply { removeAll(keys)
    }.optimizeReadOnlyMap()


// do not expose for now
@PublishedApi
internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SetsKt")
@file:OptIn(kotlin.experimental
al.ExperimentalTypeInference::class)

package kotlin.collections

import kotlin.contracts.*

internal object


/**
 * Returns an
empty read-only set.  The returned set is serializable (JVM).
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> emptySet(): Set<T> = EmptySet

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the
 * order they were specified.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> setOf(): Set<T> =
    EmptySet

/**
 * Returns an empty new 
[MutableSet].
 * @sample samples.collections.Collections.Sets.emptyMutableSet
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> hashSetOf(): HashSet<T> =
    HashSet()
public inline fun <T> linkedSetOf(vararg elements: T): LinkedHashSet<T> = LinkedHashSet(elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))).%n/**
 * Returns a new read-only set only with those given elements, that are not null.%n * Elements of the set are iterated in the order they were specified.%n * The returned set is serializable (JVM).%n */%n
@SinceKotlin("1.1")%n@kotlin.internal.InlineOnly%npublic inline fun <T> linkedSetOf(): LinkedHashSet<T> = LinkedHashSet.%n

/**
 * Returns a new [LinkedHashSet] with the given elements.%n * Elements of the set are iterated in the order they were specified.%n * Elements of the set are serializable (JVM).%n */%n
@SinceKotlin("1.4")%npublic fun <T : Any> setOfNonNull(element: T?): Set<T> = if (element != null) setOf(element) else emptySet().%n

/**
 * Returns a new read-only set only with those given elements, that are not null.%n * Elements of the set are iterated in the order they were specified.%n * The returned set is serializable (JVM).%n */%n
@SinceKotlin("1.4")%npublic fun <T : Any> setOfNullable(vararg elements: T?): Set<T> {
    return elements.filterNotNullTo(LinkedHashSet()).%n
}

/**
 * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]%n * and returning a read-only set with the same elements.%n * The set passed as a receiver to the [builderAction] is valid only inside that function.%n * Using it outside of the function produces an unspecified behavior.%n */%n
public fun <E>
    buildSet(@BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> {
contract {
    callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }%n    return buildSetInternal(builderAction)%n}

@PublishedApi
@SinceKotlin("1.3")%n@ExperimentalStdlibApi
@kotlin.internal.InlineOnly%ninternal expect inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E>

/**
 * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]%n * and returning a read-only set with the same elements.%n * The set passed as a receiver to the [builderAction] is valid only inside that function.%n * Using it outside of the function produces an unspecified behavior.%n */%n
val=x

/**
 * Returns this Set if it's not `null` and the empty set otherwise.%n */%n
@kotlin.internal.InlineOnly%npublic inline fun <T> Set<T>?.orEmpty(): Set<T> = this ?: emptySet().%n

internal fun <T> Set<T>.optimizeReadOnlySet() = when (size) {
    0 -> emptySet().%n    1 -> setOf(iterator().next()).%n    else -> this
}

/**
 * Parses the string as a signed [Byte] number and returns the result%n * or `null` if the string is not a valid representation of a number.%n */%n
@SinceKotlin("1.1")%npublic fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10).%n

/**
 * Parses the string as a signed [Byte] number and returns the result%n * or `null` if the string is not a valid representation of a number.%n */%n
@SinceKotlin("1.1")%npublic fun String.toByteOrNull(radix: Int): Byte? {
    val int = this.toIntOrNull(radix) ?:%n    return int.toByte().%n
}

/**
 * Parses the string as a [Short] number and returns the result%n * or `null` if the string is not a valid representation of a number.%n */%n
@SinceKotlin("1.1")%npublic fun String.toShortOrNull(): Short? =
toShortOrNull(radix = 10)\n\n\n* Parses the string as a [Short] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n/**\n * Parses the string as an [Int] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toShortOrNull(radix: Int): Short? {
    val int = this.toIntOrNull(radix) ?: return null
    if (int < Short.MIN_VALUE || int > Short.MAX_VALUE) return null
    return int.toShort()
}

/**\n * Parses the string as an [Int] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toIntOrNull(): Int? = toIntOrNull(radix = 10)

/**\n * Parses the string as an [Int] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toIntOrNull(radix: Int): Int? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null
    val start = 0
    val isNegative: Boolean
    val limit: Int

    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after

        start = 1

        if (firstChar == '-') {
            isNegative = true
            limit = Int.MIN_VALUE
        } else if (firstChar == '+') {
            isNegative = false
            limit = -Int.MAX_VALUE
        } else
            return null
    } else {
        start = 0
        isNegative = false
        limit = -Int.MAX_VALUE
    }

\n\n    val limitForMaxRadix = (-Int.MAX_VALUE) / 36
\n\n\n    var limitBeforeMul = limitForMaxRadix
    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
                if (result < limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }
        result *= radix
        if (result < limit + digit)
            return null
        result -= digit
    }

\n\n    return if (isNegative) result else -result
}

/**\n * Parses the string as a [Long] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toLongOrNull(): Long? = toLongOrNull(radix = 10)

/**\n * Parses the string as a [Long] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null
    val start = 0
    val isNegative: Boolean
    val limit: Long

    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after

        start = 1

        if (firstChar == '-') {
            isNegative = true
            limit = Long.MIN_VALUE
        } else if (firstChar == '+') {
            isNegative = false
            limit = -Long.MAX_VALUE
        } else
            return null
    } else {
        start = 0
        isNegative = false
        limit = -Long.MAX_VALUE
    }

\n\n    val limitForMaxRadix = (-Long.MAX_VALUE) / 36
\n\n\n    var limitBeforeMul = limitForMaxRadix
    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
                if (result < limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }
        result *= radix
        if (result < limit + digit)
            return null
        result -= digit
    }

\n\n    return if (isNegative) result else -result
}

/**\n * Parses the string as a [Double] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toDoubleOrNull(): Double? = toDoubleOrNull(radix = 10)

/**\n * Parses the string as a [Double] number and returns the result\n * or 'null' if the string is not a valid representation of a number.\n * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.\n * @SinceKotlin("1.1")\n\n\n */
@SinceKotlin("1.1")
public fun String.toDoubleOrNull(radix: Int): Double? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null
    val start = 0
    val isNegative: Boolean
    val limit: Double

    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after

        start = 1

        if (firstChar == '-') {
            isNegative = true
            limit = Double.MIN_VALUE
        } else if (firstChar == '+') {
            isNegative = false
            limit = -Double.MAX_VALUE
        } else
            return null
    } else {
        start = 0
        isNegative = false
        limit = -Double.MAX_VALUE
    }

\n\n    val limitForMaxRadix = (-Double.MAX_VALUE) / 36
\n\n\n    var limitBeforeMul = limitForMaxRadix
    var result = 0
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
                if (result < limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }
        result *= radix
        if (result < limit + digit)
            return null
        result -= digit
    }

\n\n    return if (isNegative) result else -result
}
open source used in DNAC Platform Arica 9582

Authors@PublishedApi

@PublishedApi

internal fun ulongDivide(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63
        return if (v1 < v2) ULong(0) else ULong(1)
    }
    // Optimization - use signed division if both dividend and divisor < 2^63
    if (dividend >= 0) {
        return ULong(dividend / divisor)
    }
    // Otherwise, approximate the quotient, check, and correct if necessary:
    val quotient = ((dividend ushr 1) / divisor) shl 1
    val rem = dividend - quotient * divisor
    return if (ULong(rem) >= ULong(divisor)) ULong(divisor) else 0
}

@PublishedApi

internal fun ulongRemainder(v1: ULong, v2: ULong): ULong {
    val dividend = v1.toLong()
    val divisor = v2.toLong()
    if (divisor < 0) { // i.e., divisor >= 2^63
        return if (v1 < v2) v1 else v1 - v2
    }
    // Optimization - use signed modulus if both dividend and divisor < 2^63
    if (dividend >= 0) {
        return ULong(dividend % divisor)
    }
    // Otherwise, approximate the quotient, check, and correct if necessary:
    val quotient = ((dividend ushr 1) / divisor) shl 1
    val rem = dividend - quotient * divisor
    return ULong(rem - if (ULong(rem) >= ULong(divisor)) divisor else 0)
}

@PublishedApi

internal fun doubleToUInt(v: Double): UInt = when {
    v.isNaN() -> 0u
    v <= UInt.MIN_VALUE.toDouble() -> UInt.MIN_VALUE
    v >= UInt.MAX_VALUE.toDouble() -> UInt.MAX_VALUE
    v <= Int.MAX_VALUE.toDouble() -> v.toInt().toUInt()
    else -> (v - Int.MAX_VALUE.toDouble()).toInt().toUInt() + Int.MAX_VALUE.toDouble() + 1
}

@PublishedApi

internal fun doubleToULong(v: Double): ULong = when {
    v.isNaN() -> 0u
    v <= ULong.MIN_VALUE.toDouble() -> ULong.MIN_VALUE
    v >= ULong.MAX_VALUE.toDouble() -> ULong.MAX_VALUE
    v < Long.MAX_VALUE.toDouble() -> v.toLong().toULong()
    else -> (v - 9223372036854775808.0).toLong().toULong() + 9223372036854775808uL
}

@PublishedApi

internal fun uintToDouble(v: Int): Double = (v and Int.MAX_VALUE).toDouble() + (v ushr 31 shl 30).toDouble() * 2

@PublishedApi

internal fun ulongToDouble(v: Long): Double = (v ushr 11).toDouble() * 2048 + (v and 2047)

internal fun ulongToString(v: Long): String = ulongToString(v, 10)

internal fun ulongToString(v: Long, base: Int): String {
    if (v >= 0) return v.toString(base)
    var quotient = ((v ushr 1) / base) shl 1
    var rem = v - quotient * base
    if (rem >= base) {
        rem -= base
        quotient += 1
    }
    return quotient.toString(base) + rem.toString(base)
}

package kotlin.collections

/**
 * Given an [iterator] function constructs an [Iterable] instance that returns values through
 * the [Iterator] instance provided by that function.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
    override fun iterator(): Iterator<T> = iterator()
}

internal class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
    override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())
}

/**
 * Returns true when it's safe to convert this collection to a set without changing contains method behavior.
 */
private fun <T> Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList

/** Converts this collection to a set, when it's safe to do so.
 */
internal fun <T> Iterable<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> =
    when (this) {
        is Set -> this
        else -> source
    }


else -> if (this.safeToConvertToSet()) toHashSet() else this
	} else -> toHashSet()
	


else -> if (this.safeToConvertToSet()) toHashSet() else this
	} else -> toHashSet()
	}

/**
 * Converts this collection to a set, when it's worth so and it doesn't change contains method behavior. */
internal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =
    when (this) {
        is Set -> this
        is Collection -> if (this.safeToConvertToSet()) toHashSet() else this
        else -> toHashSet()
    }

/**
 * Returns a single list of all elements from all collections in the given collection.
 */
@sample
samples.collections.Operations.flattenIterable

/**
 * @sample samples.collections.Operations.flattenIterable
 */
public fun <T> Iterable<Iterable<T>>.flatten(): List<T> {
    val result = ArrayList<T>()
    for (element in this) {
        result.addAll(element)
    }
    return result
}

/**
 * Returns a pair of lists, where
 * *first* list is built from the first values of each pair from the sequence,
 * *second* list is built from the second values of each pair from the sequence.
 */
@sample
samples.collections.Operations.unzipIterable

/**
 * Creates a sequence that returns all elements from this iterator. The sequence is constrained to be iterated only once.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()

/**
 * Creates a sequence that returns the specified values.
 */
public fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence() else

/**
 * Returns an empty sequence.
 */
public fun <T> emptySequence(): Sequence<T> = EmptySequence

private object EmptySequence : Sequence<Nothing>,
    DropTakeSequence<Nothing> {
    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
    }

/**
 * Returns a sequence if
 * it's not null and the empty sequence otherwise.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>?.orEmpty(): Sequence<T> = this ?: emptySequence()

/**
 * Returns a sequence that iterates through the elements either of
 * this sequence or, if this sequence turns out to be empty, of the sequence returned by 
 * [defaultValue] function.
 */
@SinceKotlin("1.3")
public fun <T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>): Sequence<T> = sequence {
    val iterator = this@ifEmpty.iterator()
    if (iterator.hasNext()) {
        yieldAll(iterator)
    } else {
        yieldAll(defaultValue())
    }
}

/**
 * Returns a sequence of all elements from all sequences in this
 * sequence.
 */
public fun <T> Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }

private fun <T, R> Sequence<T>.flatten(iterator: (T) -> Iterator<R>): Sequence<R> {
    if (this is TransformingSequence<*, *>)
        return (this as TransformingSequence<*,
            T>).flatten(iterator)
    return FlatteningSequence(this, { it }, iterator)
}

/**
 * Returns a pair of lists,
where

*first* list is built from the first values of each pair from this sequence,

*second* list is built from the second values of each pair from this sequence.

The operation is _terminal_ and _stateful_.

`unzip()`:

```kotlin
Pair<List<T>, List<R>>
```

`unzip()` builds two lists from the input pairs:

- *first* list is built from the first values of each pair from this sequence,
- *second* list is built from the second values of each pair from this sequence.

The operation is _terminal_.

@sample

```kotlin
samples.collections.Sequences.Transformations.unzip
```

```
open fun <T, R> Sequence<Pair<T, R>>.unzip():
Pais<List<T>, List<R>> { 
  val listT = ArrayList<T>()
  val listR = ArrayList<R>()
  for (pair in this) {
    listT.add(pair.first)
    listR.add(pair.second)
  }
  return listT to listR
}
```

**Returns a sequence that yields elements of this sequence randomly shuffled.**

Note that every iteration of the sequence returns elements in a different order.

The operation is _intermediate_ and _stateful_.

`shuffled()`:

```kotlin
Sequence<T>.shuffled(): Sequence<T> = shuffled(Random)
```

Note that every iteration of the sequence returns elements in a different order.

The operation is _intermediate_ and _stateful_.

`shuffled()` shuffles the sequence using a specified `Random` instance as the source of randomness.

```kotlin
shuffled(Random): Sequence<T> = sequence<T> { 
  val buffer = toMutableList() 
  while (buffer.isNotEmpty()) {
    val j = random.nextInt(buffer.size)
    val last = buffer.removeLast()
    val value = if (j < buffer.size) buffer.set(j, last)
      else last
    yield(value)
  }
}
```

**Returns a sequence that yields elements of this sequence randomly shuffled using the specified [random] instance as the source of randomness.**

The operation is _intermediate_ and _stateful_.

`shuffled(random: Random): Sequence<T> = sequence<T> {
  val buffer = toMutableList() 
  while (buffer.isNotEmpty()) {
    val j = random.nextInt(buffer.size)
    val last = buffer.removeLast()
    val value = if (j < buffer.size) buffer.set(j, last)
      else last
    yield(value)
  }
}
```

**Shuffles the sequence using the specified `Random` instance as the source of randomness.**

The operation is _intermediate_ and _stateful_.

`FilteringSequence<T>`:

```kotlin
FilteringSequence<T>(sequence: Sequence<T>, sendWhen: Boolean = true, predicate: (T) -> Boolean) : Sequence<T> {
  override fun iterator(): Iterator<T> = object : Iterator<T> {
    val iterator = sequence.iterator()
    var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
    var nextItem: T? = null

    private fun calcNext() {
      while (iterator.hasNext()) {
        val item = iterator.next()
        if (predicate(item) == sendWhen) {
          nextItem = item
          nextState = 1
          return
        }
      }
      nextState = 0
    }

    override fun next(): T {
      if (nextState == -1)
        calcNext()
      if (nextState == 0)
        throw NoSuchElementException()
      val result = nextItem
      nextItem = null
      nextState = -1
      return result
    }

    override fun hasNext(): Boolean {
      if (nextState == -1)
        calcNext()
      return nextState == 1
    }
  }
}
```

`FilteringSequence<T>` filters the sequence based on a predicate function:

**Returns a sequence that matches or does not match the specified [predicate].**

The operation is _intermediate_ and _stateful_.

`TransformingSequence<T, R>`:

```kotlin
TransformingSequence<T, R>(sequence: Sequence<T>, transformer: (T) -> R) : Sequence<R> {
  override fun iterator(): Iterator<R> = object : Iterator<R> {
    val iterator = sequence.iterator()
    override fun next(): R {
      return transformer(iterator.next())
    }

    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
```

`TransformingSequence<T, R>` applies a function to each value in the sequence:

**Returns a sequence that applies the given [transformer] function to the values in the underlying [sequence].**

The operation is _intermediate_.

`TransformingIndexedSequence<T, R>`:

```kotlin
TransformingIndexedSequence<T, R>(sequence: Sequence<T>, transformer: (Int, T) -> R) : Sequence<R> {
  override fun iterator(): Iterator<R> = object : Iterator<R> {
    val iterator = sequence.iterator()
    var index = 0
    override fun next(): R {
      return transformer(checkIndexOverflow(index++), iterator.next())
    }

    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
```

`TransformingIndexedSequence<T, R>` applies a function to each value and its index:

**Returns a sequence that applies the given [transformer] function to the values and their indices.**

The operation is _intermediate_.

`IndexingSequence<T>`:

```kotlin
IndexingSequence<T>(sequence: Sequence<T>) : Sequence<IndexedValue<T>> {
  override fun iterator(): Iterator<IndexedValue<T>> = object : Iterator<IndexedValue<T>> {
    val iterator = sequence.iterator()
    var index = 0
    override fun next(): IndexedValue<T> {
      return IndexedValue(checkIndexOverflow(index++), iterator.next())
    }

    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
```

`IndexingSequence<T>` returns IndexedValue objects along with the sequence:

**Returns a sequence that applies the given [transformer] function to the values and their indices.**

The operation is _intermediate_.

`FlatteningSequence<T, R>`:

```kotlin
FlatteningSequence<T, R>: Sequence<FlatteningSequence<T, R, E>>(sequence, transformer, iterator: FlatteningSequence<T, R, E>) : Sequence<E> {
  override fun iterator(): Iterator<E> = object : Iterator<E> {
    val iterator = sequence.iterator()
    val buffer = toMutableList()
    override fun next(): E {
      return transformer(iterator.next(), buffer.removeLast())
    }

    override fun hasNext(): Boolean {
      return iterator.hasNext()
    }
  }
}
```

`FlatteningSequence<T, R>` combines values from two sequences:

**Returns a sequence that combines values from two parallel underlying sequences.**

The operation is _intermediate_.

`窍聪`
function. The sequence stops returning\n * values as soon as one of the underlying sequences stops returning
values.\n */\ninternal class MergingSequence<T1, T2, V>\nconstructor(\n private val sequence1:
Sequence<T1>,\n private val sequence2: Sequence<T2>,\n private val transform: (T1, T2) -> V\n) :
Sequence<V> {\n override fun iterator(): Iterator<V> = object : Iterator<V> {\n
val iterator1 =
sequence1.iterator()\n
val iterator2 = sequence2.iterator()\n
override fun next(): V {\n
return
transform(iterator1.next(), iterator2.next())\n
}\n\n
override fun hasNext(): Boolean {\n
return
iterator1.hasNext() && iterator2.hasNext()\n
}\n }\n}\n\ninternal class FlatteningSequence<T, R,
E>\nconstructor(\n private val sequence: Sequence<T>,\n private val transformer: (T) -> R,\n private val
iterator: (R) -> Iterator<E>\n) : Sequence<E> {\n override fun iterator(): Iterator<E> = object : Iterator<E> {\n
val iterator = sequence.iterator()\n
var itemIterator: Iterator<E>? = null\n\n
override fun next(): E {\n
if (!ensureItemIterator())\n
throw NoSuchElementException()\n
return itemIterator!!.next()\n
}\n\n
override fun hasNext(): Boolean {\n
return ensureItemIterator()\n
}\n\n
private fun
ensureItemIterator(): Boolean {\n
if (itemIterator?.hasNext() == false)\n
itemIterator = null\n\n
while (itemIterator == null) {\n
if (!iterator.hasNext()) {\n
return false\n
} else {\n
val element = iterator.next()\n
val nextItemIterator = iterator(transformer(element))\n
if (nextItemIterator.hasNext()) {\n
itemIterator = nextItemIterator\n
return true\n
}\n
}\n
}\n
return true\n
}\n }\n}\n\ninternal fun <T, C, R> flatMapIndexed(source:
Sequence<T>, transform: (Int, T) -> C, iterator: (C) -> Iterator<R>): Sequence<R> =\n sequence {\n
var
index = 0\n
for (element in source) {\n
val result = transform(checkIndexOverflow(index++), element)\n
yieldAll(iterator(result))\n
}\n }\n\n/**\n * A sequence that supports drop(n) and take(n) operations\n
*/\ninternal interface DropTakeSequence<T> : Sequence<T> {\n fun drop(n: Int): Sequence<T>\n fun take(n:
Int): Sequence<T>\n}\n\n/**\n * A sequence that skips [startIndex] values from the underlying [sequence]\n * and
stops returning values right before [endIndex], i.e. stops at `endIndex - 1`\n */\ninternal class SubSequence<T>(\n
private val sequence: Sequence<T>,\n private val startIndex: Int,\n private val endIndex: Int\n) : Sequence<T>,
DropTakeSequence<T> {\n\n init {\n
require(startIndex >= 0) { \"startIndex should be non-negative, but is
$startIndex\" }\n
require(endIndex >= 0) { \"endIndex should be non-negative, but is $endIndex\" }\n
require(endIndex >= startIndex) { \"endIndex should be not less than startIndex, but was $endIndex < $startIndex\"
}\n }\n\n private val count: Int get() = endIndex - startIndex\n\n override fun drop(n: Int): Sequence<T> = if (n
>= count) emptySequence() else SubSequence(sequence, startIndex + n, endIndex)\n override fun take(n: Int):
Sequence<T> = if (n >= count) this else SubSequence(sequence, startIndex, startIndex + n)\n\n override fun
iterator() = object : Iterator<T> {\n\n
val iterator = sequence.iterator()\n
var position = 0\n\n
//
Shouldn't be called from constructor to avoid premature iteration\n
private fun drop() {\n
while (position
< startIndex && iterator.hasNext()) {\n
iterator.next()\n
position++\n
}\n
}\n\n
override fun hasNext(): Boolean {\n
drop()\n
return (position < endIndex) && iterator.hasNext()\n
}\n\n
override fun next(): T {\n
drop()\n
if (position >= endIndex)\n
throw
NoSuchElementException()\n
position++\n
return iterator.next()\n
}\n }\n}\n\n/**\n * A
sequence that returns at most [count] values from the underlying [sequence], and stops returning values\n * as soon
as that count is reached.\n */\ninternal class TakeSequence<T>(\n private val sequence: Sequence<T>,\n private
val count: Int\n) : Sequence<T>, DropTakeSequence<T> {\n\n init {\n
require(count >= 0) { \"count must be
non-negative, but was $count.\" }\n }\n\n override fun drop(n: Int): Sequence<T> = if (n >= count)
emptySequence() else SubSequence(sequence, n, count)\n override fun take(n: Int): Sequence<T> = if (n >=
count) this else TakeSequence(sequence, n)\n\n override fun iterator(): Iterator<T> = object : Iterator<T> {\n
var left = count\n
val iterator = sequence.iterator()\n\n
override fun next(): T {\n
if (left == 0)\n
throw NoSuchElementException()\n
left--\n
return iterator.next()\n
}\n\n
override fun
hasNext(): Boolean {\n
return left > 0 && iterator.hasNext()\n
}\n }\n}\n\n/**\n * A sequence that
returns values from the underlying [sequence] while the [predicate] function returns\n * `true`, and stops returning
values once the function returns `false` for the next element.\n */\ninternal class
TakeWhileSequence<T>\nconstructor(\n private val sequence: Sequence<T>,\n private val predicate: (T) ->

Open Source Used In DNAC Platform Arica 9585


Override fun iterator(): Iterator<T> = object : Iterator<T> {
    override fun hasNext(): Boolean = nextState != -1
    override fun next(): T {
        if (nextState == -1)
            calcNext() // will change nextState
        return nextState == 1
    }
}

/**
 * Returns a wrapper sequence that provides
 * the values after that.
 */
private class GeneratorSequence<T : Any>(private val
    source: Sequence<T>, private val keySelector: (T) -> K) : Sequence<T> {
    override fun iterator(): Iterator<T> = object : Iterator<T> {
        var nextItem: T? = null
        var nextState: Int = -2 // -2 for initial
        override fun computeNext(): T {
            if (nextState == -1)
                calcNext() // will change nextState
            return nextState == 1
        }
    }
}

/**
 * A sequence that skips the specified number of values from the underlying 
 * [sequence] and returns all values after that.
 */
internal class DropSequence<T>(
    private val
    sequence: Sequence<T>,
    private val count: Int
) : Sequence<T>, DropTakeSequence<T> {
    init {
        @Suppress("UNCHECKED_CAST")
        val result = nextItem as T
        nextItem = null
        nextState = -1
        return result
    }

    override fun drop(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) DropSequence(this, n) else DropSequence(sequence, n) }
values of this sequence, but ensures it can be iterated only one time.

* The operation is _intermediate_ and _stateless_.

* `IllegalStateException` is thrown on iterating the returned sequence from the second time.

* `ConstrainedOnceSequence<T>` returns a sequence which invokes the function to calculate the next value on each iteration until the function returns `null`. The returned sequence is constrained to be iterated only once.

* `generateSequence(seed: T?, nextFunction: (T) -> T?): Sequence<T>` returns a sequence defined by the starting value `seed` and the function `nextFunction`, which is invoked to calculate the next value based on the previous one on each iteration. If `seed` is `null`, an empty sequence is produced. The sequence can be iterated multiple times, each time starting with `seed`.

* `generateSequence(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T>` returns a sequence defined by the function `seedFunction`, which is invoked to produce the starting value, and the `nextFunction`, which is invoked to calculate the next value based on the previous one. If `seedFunction` returns `null`, an empty sequence is produced.

* `generateSequenceWithSeed(seed: T?, nextFunction: (T) -> T?): Sequence<T>` returns a sequence defined by the starting value `seed` and the function `nextFunction`, which is invoked to calculate the next value based on the previous one. If `seed` is `null`, an empty sequence is produced. The sequence can be iterated multiple times, each time starting with `seed`.

* `generateSequenceWithLazySeed(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T>` returns a sequence defined by the function `seedFunction`, which is invoked to produce the starting value, and the `nextFunction`, which is invoked to calculate the next value based on the previous one. If `seedFunction` returns `null`, an empty sequence is produced.

* `require(value: Boolean): Unit` throws an `IllegalArgumentException` if the `value` is false.

* `require(value: Boolean, lazyMessage: () -> Any): Unit` throws an `IllegalArgumentException` with the result of calling `lazyMessage` if the `value` is false.

* `requireNotNull(value: T?): T` throws an `IllegalArgumentException` if the `value` is null. Otherwise returns the not null value.

* `check(value: Boolean): Unit` throws an `IllegalStateException` if the `value` is false.
if (!value) {
    val message = lazyMessage()
    throw IllegalStateException(message.toString())
}

/**
 * Throws an [IllegalStateException] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [lazyMessage] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, message: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(message)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        throw IllegalStateException(errorMessage)
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the result of calling [message] if the [value] is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> checkNotNull(value: T?, errorMessage: () -> String): T {
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = errorMessage()
[index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

```kotlin
public actual fun CharArray.elementAt(index: Int): Char
{
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size}"
}
}

/**
 * Returns a [List] that wraps the original array.
 * @sample
 * samples.collections.Collections.Elements.elementAt
 */
public actual fun CharArray.asList(): List<Char> {
    return object : AbstractList<Char>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: Char): Boolean = this@asList.contains(element)
        override fun get(index: Int): Char {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: Char): Int {
            if ((element as Any?) !is Char) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: Char): Int {
            if ((element as Any?) !is Char) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
```

**Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too.**

```kotlin
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

**Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too.**

```kotlin
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *deeply* equal to one another, i.e. contain the same number of the same elements in the same order. If two corresponding elements are nested arrays, they are also compared deeply.
 * If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public actual fun <T, U> Array<out T>.contentDeepEquals(other: Array<out U>): Boolean {
    return this.contentDeepEquals(other)
}
```

```kotlin
/**
 * Returns `true` if the two specified arrays are *deeply* equal to one another, i.e. contain the same number of the same elements in the same order. If two corresponding elements are nested arrays, they are also compared deeply.
 * If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

```kotlin
/**
 * Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

```kotlin
/**
 * Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

```kotlin
/**
 * Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```

```kotlin
/**
 * Returns a hash code based on the contents of this array as if it is [List]. Nested arrays are treated as lists too. If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.4")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}
```
T>?.contentDeepHashCode(): Int {
    return this.contentDeepHashCode()
}

/**
 * Returns a string representation of the contents of this array as if it is a [List].
 * Nested arrays are treated as lists too.
 * If any of arrays contains itself on any nesting level that reference
 * is rendered as "\[...\]" to prevent recursion.
 * @sample samples.collections.Arrays.ContentOperations.contentDeepToString
 */
@SinceKotlin("1.1")
@kotlin.internal.LowPriorityInOverloadResolution
public actual fun <T> Array<T>.contentDeepToString(): String {
    return this.contentDeepToString()
}

/**
 * Returns a string representation of the contents of this array as if it is a [List].
 * Nested arrays are treated as lists too.
 * If any of arrays contains itself on any nesting level that reference
 * is rendered as "\[...\]" to prevent recursion.
 * @sample samples.collections.Arrays.ContentOperations.contentDeepToString
 */
@SinceKotlin("1.4")
@library("arrayDeepToString")
public actual fun <T> Array<T>?.contentDeepToString(): String {
    return this.contentDeepToString()
}

/**
 * Returns `true` if the two specified arrays
 * are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun ShortArray.contentEquals(other: ShortArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun IntArray.contentEquals(other: IntArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun LongArray.contentEquals(other: LongArray): Boolean {
    return this.contentEquals(other)
}

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 * @sample samples.collections.Arrays.ContentOperations.contentEquals
 * @deprecated Use Kotlin compiler 1.4 to avoid deprecation warning.
 */
@DeprecatedSinceKotlin(hiddenSince = "1.4")
public actual infix fun FloatArray.contentEquals(other: FloatArray): Boolean {
    return this.contentEquals(other)
}
function.

\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n\@SinceKotlin("1.1\")\n\@DeprecatedSinceKotlin(hidenSince = "1.4")\npublication actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean {\n\treturn this.contentEquals(other)\n}\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\n\@SinceKotlin("1.1\")\n\@DeprecatedSinceKotlin(hidenSince = "1.4")\npublication actual infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean {\n\treturn this.contentEquals(other)\n}\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun CharArray.contentEquals(other: CharArray): Boolean {\n\treturn this.contentEquals(other)\n}\n\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun ByteArray?.contentEquals(other: ByteArray?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun ShortArray?.contentEquals(other: ShortArray?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun IntArray?.contentEquals(other: IntArray?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun LongArray?.contentEquals(other: LongArray?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it means that `\'NaN\'` is equal to itself and `-0.0` is not equal to `-0.0`.

\*\*@SinceKotlin("1.4")\n\@library("arrayEquals")\npublication actual infix fun FloatArray?.contentEquals(other: FloatArray?): Boolean {\n\tdefinedExternally\n\n\* Returns `true` if the two specified arrays are structurally equal to one another,\n\* i.e. contain the same number of the same elements in the same order.\n\* The elements are compared for equality with the [equals][Any.equals] function.\n\* For floating point numbers it
means that 'NaN' is equal to itself and '-'0.0' is not equal to '0.0'.

*\@SinceKotlin("1.4\")\@library("arrayEquals")\*public actual infix fun DoubleArray?.contentEquals(other: DoubleArray?): Boolean (\* definedExternally\*)\n\n* \* Returns 'true' if the two specified arrays are structurally* equal to one another;\n\* i.e. contain the same number of the same elements in the same order.\n* \* The elements are compared for equality with the [equals][Any.equals] function.\n* For floating point numbers it means that 'NaN' is equal to itself and '-'0.0' is not equal to '0.0'.\n
*\@SinceKotlin("1.4\")\@library("arrayEquals")\*public actual infix fun BooleanArray?.contentEquals(other: BooleanArray?): Boolean (\* definedExternally\*)\n\n* \* Returns 'true' if the two specified arrays are structurally* equal to one another;\n\* i.e. contain the same number of the same elements in the same order.\n* \* The elements are compared for equality with the [equals][Any.equals] function.\n* For floating point numbers it means that 'NaN' is equal to itself and '-'0.0' is not equal to '0.0'.\n
*\@SinceKotlin("1.4\")\@library("arrayEquals")\*public actual infix fun CharArray?.contentEquals(other: CharArray?): Boolean (\* definedExternally\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun \n\n\nArray<out T>.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun ByteArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun ShortArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun IntArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun LongArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun FloatArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun DoubleArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun IntArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun CharArray.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun Array<out T>.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun ByteArray?.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun ShortArray?.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun IntArray?.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun CharArray?.contentHashCode(): Int (\* definedHashCode()\*)\n\n* \* Returns a hash code based on the contents of this array as if it is [List]\.*\@Deprecated("Use Kotlin compiler 1.4 to avoid deprecation warning.\")\@SinceKotlin("1.1\")\@DeprecratedSinceKotlin(hiddenSince = "1.4\")\*public actual fun
of this array as if it is [List].

```kotlin
/**
 * Returns a hash code based on the
 * contents of this array as if it is [List].
 */
@SinceKotlin("1.4")
@library("arrayHashCode")
public actual fun LongArray?.contentHashCode(): Int {
    definedExternally
}

/**
 * Returns a hash code based on the
 * contents of this array as if it is [List].
 */
@SinceKotlin("1.4")
@library("arrayHashCode")
public actual fun FloatArray?.contentHashCode(): Int {
    definedExternally
}

/**
 * Returns a hash code based on the
 * contents of this array as if it is [List].
 */
@SinceKotlin("1.4")
@library("arrayHashCode")
public actual fun DoubleArray?.contentHashCode(): Int {
    definedExternally
}

/**
 * Returns a hash code based on the
 * contents of this array as if it is [List].
 */
@SinceKotlin("1.4")
@library("arrayHashCode")
public actual fun BooleanArray?.contentHashCode(): Int {
    definedExternally
}

/**
 * Returns a hash code based on the
 * contents of this array as if it is [List].
 */
@SinceKotlin("1.4")
@library("arrayHashCode")
public actual fun CharArray?.contentHashCode(): Int {
    definedExternally
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun <T> Array<out T>.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun ByteArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun ShortArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun IntArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun LongArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun FloatArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun DoubleArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun BooleanArray.contentToString(): String {
    return this.contentToString()
}

/**
 * Returns a string representation of the
 * contents of the specified array as if it is [List].
 */
@Deprecated("Use Kotlin compiler 1.4 to
avoid deprecation warning.")
@SinceKotlin("1.1")
public actual fun CharArray.contentToString(): String {
    return this.contentToString()
}
```

---

**Open Source Used in DNAC Platform Arica 9593**
/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun ByteArray?.contentToString(): String{
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun ShortArray?.contentToString(): String{
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun IntArray?.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun LongArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun FloatArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is 
 * [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun DoubleArray?.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as 
 * if it is [List].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.4")
@library("arrayToString")
public actual fun BooleanArray?.contentToString(): String

/**
 * Copies this array or its subrange into the [destination] array and returns that 
 * array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the 
 * subrange so that it overlaps with the destination range.
 * 
 * @param destination the array to copy to
 * @param startIndex the beginning (inclusive) of the subrange to copy. By default, it
 * is 0. 
 * @param endIndex the end (exclusive) of the subrange to copy, size of this 
 * array by default. By default, it
 * is 0.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when 
 * startIndex
 * > endIndex.
 
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * 
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the 
 * [destination] array starting at the specified 
 * [destinationOffset].
 
 * or when that index is out of the [destination] array indices range. 
 * 
 * @return the 
 * [destination] array.

 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)
    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)
    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)
    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)
    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun IntArray_copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun LongArray_copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun ByteArray_copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun ShortArray_copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun IntArray_copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/

/*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun LongArray_copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size)\n    return this.copyOf(selector = CopyIntoSelector(destination, destinationOffset, startIndex, endIndex), size = size)\n*/
IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. \n * @param destination the array to copy to. \n * @param destinationOffset the position in the [destination] array to copy to. \n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default. \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. \n * @return the [destination] array.

*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS!)")
public actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = shape.size): FloatArray {
    arrayCopy(this.unsafeCast<Array<Float>>(), destination.unsafeCast<Array<Float>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. \n * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to. \n * @param startIndex the beginning (inclusive) of the subrange to copy, size of this array by default. \n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default. \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. \n * @return the [destination] array.

*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS!)")
public actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = shape.size): DoubleArray {
    arrayCopy(this.unsafeCast<Array<Double>>(), destination.unsafeCast<Array<Double>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. \n * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to. \n * @param startIndex the beginning (inclusive) of the subrange to copy, size of this array by default. \n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default. \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. \n * @return the [destination] array.

*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS!)")
public actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = shape.size): BooleanArray {
    arrayCopy(this.unsafeCast<Array<Boolean>>(), destination.unsafeCast<Array<Boolean>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. \n * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to. \n * @param startIndex the beginning (inclusive) of the subrange to copy, size of this array by default. \n * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default. \n * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. \n * @return the [destination] array.

*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS!)")
public actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)
    return destination
}

Open Source Used In DNAC Platform Arica 9596
* Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyOf

public actual inline fun <T> Array<out T>.copyOf(): Array<T> {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyOf

public actual inline fun ByteArray.copyOf(): ByteArray {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual inline fun ShortArray.copyOf(): ShortArray {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual inline fun IntArray.copyOf(): IntArray {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun LongArray.copyOf(): LongArray {
    return withType("LongArray", this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun FloatArray.copyOf(): FloatArray {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun DoubleArray.copyOf(): DoubleArray {
    return this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun BooleanArray.copyOf(): BooleanArray {
    return withType("BooleanArray", this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun CharArray.copyOf(): CharArray {
    return withType("CharArray", this.asDynamic().slice()!!.asDynamic().slice()!! * Returns new array which is a copy of the original array.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun ByteArray.copyOf(newSize: Int): ByteArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ByteArray(newSize))!!.asDynamic().slice()!! * Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun ShortArray.copyOf(newSize: Int): ShortArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, ShortArray(newSize))!!.asDynamic().slice()!! * Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith

public actual fun IntArray.copyOf(newSize: Int): IntArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, IntArray(newSize))!!.asDynamic().slice()!! * Returns new array which is a copy of the original array, resized to the given [newSize].
* The copy is either truncated or padded at the end with zero values if necessary.
* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
* @sample
samples.collections.Arrays.CopyOfOperations.copyWith
values.

```kotlin
* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

@public actual fun LongArray.copyOf(newSize: Int): LongArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return withType("LongArray", arrayCopyResize(this, newSize, 0L))
}

@public actual fun FloatArray.copyOf(newSize: Int): FloatArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, FloatArray(newSize))
}

@public actual fun DoubleArray.copyOf(newSize: Int): DoubleArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return fillFrom(this, DoubleArray(newSize))
}

@public actual fun BooleanArray.copyOf(newSize: Int): BooleanArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return withType("BooleanArray", arrayCopyResize(this, newSize, false))
}

@public actual fun CharArray.copyOf(newSize: Int): CharArray {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return withType("CharArray", fillFrom(this, CharArray(newSize)))
}

@public actual fun <T> Array<out T>.copyOf(newSize: Int): Array<T?> {
    require(newSize >= 0) { "Invalid new array size: $newSize." }
    return arrayCopyResize(this, newSize, /*null*/)
}

@public actual fun <T> Array<out T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    return this.asDynamic().slice(fromIndex, toIndex)
}
```

* @Suppress("ACTUAL_WITHOUT_EXPECT")

@Suppress("ACTUAL_WITHOUT_EXPECT")
ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("LongArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return this.asDynamic().slice(fromIndex, toIndex)/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("BooleanArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Returns a new array which is a copy of the specified range of the original array.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*/

abstract fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size) return withType("CharArray", this.asDynamic().slice(fromIndex, toIndex))/*
 * Fills this array or its subrange with the specified [element] value.*/

*Throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.*/

*Throws IllegalArgumentException if [fromIndex] is greater than [toIndex].*
@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```

*\@SinceKotlin("1.3")*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n
```
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n    \n    *param fromIndex the start of the range (inclusive) to fill, 0 by default.\n    \n    *param toIndex the end of the range (exclusive) to fill, size of this array by default.\n    \n    *param element the value to fill the array with.\n    \n    \@throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n    \@throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
```
than [toIntIndex].

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgument if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
    return
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgument if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
    return
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun <T> Array<out T>.plus(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(element: Byte): ByteArray {
    return plus(byteArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ShortArray.plus(element: Short): ShortArray {
    return plus(shortArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun IntArray.plus(element: Int): IntArray {
    return plus(intArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun LongArray.plus(element: Long): LongArray {
    return plus(longArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun FloatArray.plus(element: Float): FloatArray {
    return plus(floatArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun DoubleArray.plus(element: Double): DoubleArray {
    return plus(doubleArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun BooleanArray.plus(element: Boolean): BooleanArray {
    return plus(booleanArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun CharArray.plus(element: Char): CharArray {
    return plus(charArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual operator fun <T> Array<out T>.plus(elements: Collection<T>): Array<T> {
    return arrayPlusCollection(this, elements)
}
```

actual operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun ShortArray.plus(elements: Collection<Short>): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun IntArray.plus(elements: Collection<Int>): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun LongArray.plus(elements: Collection<Long>): LongArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
} *//

public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
} *//

public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
} *//

public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
} *//
CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun IntArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun LongArray.sort(): Unit {
    if (size > 1) sort {
        a: Long, b: Long -> a.compareTo(b)
    }
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun ByteArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun ShortArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun FloatArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun CharArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place according to the natural order
 * of its elements. It is _stable_. It means that equal elements preserve their order relative
to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortArrayOfComparable
 */
public actual fun <T : Comparable<T>> Array<out T>.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * The sort is _stable_. It means that equal elements preserve their order relative
to each other after sorting.
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public fun <T> Array<out T>.sort(comparison: (a: T, b: T) -> Int): Unit {
    if (size > 1) sortArrayWith(this, comparison)
}

/**
 * Sorts a range in the array in-place.
 * The sort is _stable_. It means that equal elements preserve their order relative
to each other after sorting.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentOutOfRangeException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun <T : Comparable<T>> Array<out T>.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    sortArrayWith(this, fromIndex, toIndex, naturalOrder())
    definedExternally
}
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ShortArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val subarray =
    this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<ShortArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun IntArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val subarray =
    this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<IntArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun LongArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    sortArrayWith(this.unsafeCast<Array<Long>>(),
    fromIndex, toIndex, naturalOrder())
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun FloatArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val subarray =
    this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<FloatArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun DoubleArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val subarray =
    this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<DoubleArray>()
    subarray.sort()
}

/**
 * Sorts a range in the array in-place.
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.
 * @param toIndex the end of the range (exclusive) to sort, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 * @sample samples.collections.Arrays.Sorting.sortRangeOfArray
 */
@SinceKotlin("1.4")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.sort(fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    val subarray =
    this.asDynamic().subarray(fromIndex, toIndex).unsafeCast<CharArray>()
    subarray.sort()
array in-place according to the order specified by the given [comparison] function.

Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun ShortArray.sort(noinline comparison: (a: Short, b: Short) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun IntArray.sort(noinline comparison: (a: Int, b: Int) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun LongArray.sort(noinline comparison: (a: Long, b: Long) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun FloatArray.sort(noinline comparison: (a: Float, b: Float) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun DoubleArray.sort(noinline comparison: (a: Double, b: Double) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

dynamic fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {\n    asDynamic().sort(comparison)\n}/*/un /** * Sorts the array in-place according to the order specified by the given [comparison] function.\n*/un@kotlin.internal.InlineOnly

// Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

@file:kotlin.jvm.JvmName("ComparisonsKt")
@file:kotlin.jvm.JvmMultifileClass

package
kotlin.comparisons

/* Compares two values using the specified functions [selectors] to calculate the result of the comparison.
 * The functions are called sequentially, receive the given values [a] and [b] and return [Comparable] objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do not compare as equal, the result of that comparison is returned.
 */

@sample samples.comparisons

/**
 * Compares two values using the specified [selector] function to calculate the result of the comparison.
 * The result of comparison of these [Comparable] instances is returned.
 */

@kotlin.internal.InlineOnly
public fun <T> compareValues(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValuesBy(a, b, selector)
}

/**
 * Creates a comparator using the sequence of functions to calculate a result of comparison.
 * The functions are called sequentially, receive the given values `a` and `b` and return [Comparable] objects. As soon as the [Comparable] instances returned by a function for `a` and `b` values do not compare as equal, the result of that comparison is returned from the [Comparator].
 */

@kotlin.internal.InlineOnly
public fun <T> compareBy(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}

/**
 * Creates a comparator using the sequence of functions to calculate a result of comparison.
 * The functions are called sequentially, receive the given values `a` and `b` and return [Comparable] objects. As soon as the [Comparable] instances returned by a function for `a` and `b` values do not compare as equal, the result of that comparison is returned from the [Comparator].
 */

@kotlin.internal.InlineOnly
public fun <T> compareByDescending(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    return Comparator { a, b ->
        compareValuesByDescendingImpl(a, b, selectors)
    }
}
samples.comparisons.Comparisons.compareByDescendingWithComparator

*/
@kotlin.internal.InlineOnly
public inline fun <T, K> Comparator<T>.thenBy(crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenBy.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(a, b, selector)
    }

/**
 * Creates a descending comparator using the primary comparator and the 
 * selector function to transform values and then compares them with the given 
 * comparator.
 */
@kotlin.internal.InlineOnly
public inline fun <T, K> Comparator<T>.thenByDescending(crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenByDescending.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(b, a, selector)
    }

/**
 * Creates a comparator using the primary comparator and function to calculate a result of comparison.
 */
@kotlin.internal.InlineOnly
public fun <T : Any> nullsFirst(comparator: Comparator<in T>): Comparator<T?> =
    Comparator { a, b ->
        when {
            a === b -> 0
            a == null -> -1
            b == null -> 1
            else -> comparator.compare(a, b)
        }
    }

/**
 * Provides a comparator of nullable [Comparable] values * considering `null` value less than any other value.
 */
@kotlin.internal.InlineOnly
public inline fun <T : Comparable<T>> nullsFirst(): Comparator<T?> =
    Comparator { a, b ->
        when {
            a === b -> 0
            a == null -> -1
            b == null -> 1
            else -> a.compareTo(b)
        }
    }

// Not so useful without type
inference for receiver of expression
/**
 * Extends the given [comparator] of non-nullable values to a comparator
 * of nullable values considering `null` value less than any other value.
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> nullsFirstLastWithComparator(*@sample samples.comparisons.Comparisons.nullsFirstLastWithComparator

/* */
Open Source Used In DNAC Platform Arica 9607
nullsFirst(naturalOrder())\n\n* Extends the given [comparator] of non-nullable values to a comparator of null values
  * considering `null` value greater than any other value.
* @sample samples.comparisons.Comparisons.nullsFirstWithComparator
* \n\npublic fun \n\n* Returns a comparator that compares [Comparable] objects in natural order.
* @sample samples.comparisons.Comparisons.naturalOrderComparator
* \n\n\n\n
private class
ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {
    override fun compare(a: T, b: T): Int = comparator.compare(b, a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<T> = comparator
}

private object NaturalOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = a.compareTo(b)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator
}

private object ReverseOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = b.compareTo(a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator
}

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StandardKt")
package kotlin

import kotlin.contracts.*

/**
 * An exception is thrown to indicate that a method body remains to be implemented.
 */
public class NotImplementedError(message: String = "An operation is not implemented.") : Error(message)

/**
 * Always throws [NotImplementedError] stating that operation is not implemented.
 * @param reason a string
 * explaining why the implementation is missing.
 */
@kotlin.internal.InlineOnly
public inline fun TODO(reason: String): Nothing = throw NotImplementedError("An operation is not implemented: $reason")

/**
 * Calls the specified function [block] and returns its result.
 * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).
 */
@kotlin.internal.InlineOnly
public inline fun <R> run(block: () -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    return block()
}

/**
 * Calls the specified function [block] with the given [receiver] as its receiver and returns its result.
 * For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).
 */
@kotlin.internal.InlineOnly
public inline fun <T, R> T.run(block: T.() -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    return block()
}
see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#with).

```kotlin
// Calls the specified function [block] with `this` value as its receiver and returns `this` value.
@kotlin.internal.InlineOnly
public inline fun <T, R> with(receiver: T, block: T.() -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } return receiver.block()
}
```

```kotlin
/**
 * Calls the specified function [block] with `this` value as its argument and returns `this` value.
 */
@kotlin.internal.InlineOnly
public inline fun <T> T.apply(block: T.() -> Unit): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } block() return this
}
```

```kotlin
/**
 * Calls the specified function [block] with `this` value as its argument and returns its result.
 */
@kotlin.internal.InlineOnly
@SinceKotlin("1.1")
public inline fun <T> T.also(block: (T) -> Unit): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } block(this) return this
}
```

```kotlin
/**
 * Calls the specified function [block] with `this` value as its argument and returns its result.
 */
@kotlin.internal.InlineOnly
public inline fun <T, R> T.let(block: (T) -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    } return block(this)
}
```

```kotlin
/**
 * Returns `this` value if it satisfies the given [predicate] or `null`, if it doesn't.
 */
@kotlin.internal.InlineOnly
@SinceKotlin("1.1")
public inline fun <T> T.takeIf(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)
    } return if (predicate(this)) this else null
}
```

```kotlin
/**
 * Returns `this` value if it _does not_ satisfy the given [predicate] or `null`, if it does.
 */
@kotlin.internal.InlineOnly
@SinceKotlin("1.1")
public inline fun <T> T.takeUnless(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)
    } return if (!predicate(this)) this else null
}
```

```kotlin
/**
 * Executes the given function [action] specified number of [times].
 */
@kotlin.internal.InlineOnly
public inline fun repeat(times: Int, action: (Int) -> Unit) {
    contract { callsInPlace(action) }

    for (index in 0 until times) {
        action(index)
    }
}
```

```kotlin
// NOTE: THIS FILE IS AUTO-GENERATED by the

import kotlin.js.*

/**
 * Returns the greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}
```

```kotlin
/**
 * Returns the greater of two values.
 * If either value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual inline fun <T> T.maxOf(b: T): T {
    (T) -> Unit {
        return if (a >= b) a else b
    }
}
```
/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Double, b: Double): Double {
    return Math.max(a, b)
}

/**
 * Returns the greater of three values.
 * 
 * If there are multiple equal maximal values, returns the first of them.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte {
    return Math.max(a.toInt(), b.toInt(), c.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the greater of three values.
 * 
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual
inline fun maxOf(a: Int, b: Int, c: Int): Int {
    return Math.max(a, b, c)
}

/**
 * Returns the greater of three values.
 * 
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual
fun maxOf(a: Long, vararg other: Long): Long {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 * 
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual
fun maxOf(a: Float, vararg other: Float): Float {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the greater of three values.
 * 
 * If any value is `NaN`, returns `NaN`.
 */
@SinceKotlin("1.1")
public actual
fun maxOf(a: Double, vararg other: Double): Double {
    var max = a
    for (e in other) max = maxOf(max, e)
    return max
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Byte, b: Byte): Byte {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Int, b: Int): Int {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Short, b: Short): Short {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Long, b: Long): Long {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public actual
fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
}
MIN_VALUE: ULong = ULong(0)
/**
 * A constant holding the maximum value an instance of ULong can have.
 * @since 1.0
 */
public const val MAX_VALUE: ULong = ULong(-1)
/**
 * The number of bytes used to represent an instance of ULong in a binary form.
 */
public const val SIZE_BYTES: Int = 8
/**
 * The number of bits used to represent an instance of ULong in a binary form.
 */
public const val SIZE_BITS: Int = 64

/**
 * A constant holding the maximum value an instance of ULong can have.
 */
public const val MAX_VALUE: ULong = ULong(-1)
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
 */
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toULong())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
 */
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toULong())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
 */
public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toULong())
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
 */
public inline operator fun compareTo(other: ULong): Int = this.compareTo(other.toULong())

/**
 * Adds the other value to this value.
 */
public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())
/**
 * Adds the other value to this value.
 */
public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())
/**
 * Adds the other value to this value.
 */
public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())
/**
 * Adds the other value to this value.
 */
public inline operator fun plus(other: ULong): ULong = ULong(this.data.plus(other.data))

/**
 * Subtracts the other value from this value.
 */
public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())
/**
 * Subtracts the other value from this value.
 */
public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())
/**
 * Subtracts the other value from this value.
 */
public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())
/**
 * Subtracts the other value from this value.
 */
public inline operator fun minus(other: ULong): ULong = ULong(this.data.minus(other.data))

/**
 * Multiplies this value by the other value.
 */
public inline operator fun times(other: UByte): ULong = this.times(other.toULong())
/**
 * Multiplies this value by the other value.
 */
public inline operator fun times(other: UShort): ULong = this.times(other.toULong())
/**
 * Multiplies this value by the other value.
 */
public inline operator fun times(other: UInt): ULong = this.times(other.toULong())
/**
 * Multiplies this value by the other value.
 */
public inline operator fun times(other: ULong): ULong = ULong(this.data.times(other.data))

/**
 * Divides this value by the other value.
 */
public inline operator fun div(other: UByte): ULong = this.div(other.toULong())
/**
 * Divides this value by the other value.
 */
public inline operator fun div(other: UShort): ULong = this.div(other.toULong())
/**
 * Divides this value by the other value.
 */
public inline operator fun div(other: UInt): ULong = this.div(other.toULong())
/**
 * Divides this value by the other value.
 */
public inline operator fun div(other: ULong): ULong = ULong(this.data.div(other.data))

/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(other: ULong): ULong = this.rem(other.toULong())

/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(dividend: UByte, divisor: UByte): Int = this.rem(dividend.toULong(), divisor.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(dividend: UShort, divisor: UShort): Int = this.rem(dividend.toULong(), divisor.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(dividend: UInt, divisor: UInt): Int = this.rem(dividend.toULong(), divisor.toULong())
/**
 * Calculates the remainder of dividing this value by the other value.
 */
public inline operator fun rem(dividend: ULong, divisor: ULong): Int = this/rem(dividend.toULong(), divisor.toULong())
inline operator fun rem(other: ULong): ULong = this.rem(other.toULong())

/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = ulongRemainder(this, other)

/** Increments this value. */

@kotlin.internal.InlineOnly
public inline operator fun inc(): ULong = ULong(data.inc())

/** Increments this value. */

@kotlin.internal.InlineOnly
public inline operator fun dec(): ULong = ULong(data.dec())

/** Creates a range from this value to the specified [other] value. */

@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)

/** Shifts this value left by the [bitCount] number of bits. */

@kotlin.internal.InlineOnly
public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)

/** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */

@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)

/** Performs a bitwise AND operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong = ULong(data and other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)

/** Performs a bitwise XOR operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline operator fun or(other: ULong): ULong = ULong(this.data or other.data)

/** pang a set number of bits. */

@kotlin.internal.InlineOnly
public inline operator fun unshiftedOr(bits: Int): ULong = ULong(data or ULong(1).shl(bits))

/** Performs a bitwise AND operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong = ULong(data and other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)

/** Performs a bitwise XOR operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline operator fun or(other: ULong): ULong = ULong(this.data or other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline operator fun or(other: ULong): ULong = this.toULong().or(other.toULong())

/** Returns this value. */

@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = this

/** Returns this value. */

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()

/** Returns this value. */

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Returns this value. */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Returns this value. */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [ULong] value to [Byte]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [ULong] value to [UInt]. */

@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [ULong] value to [UShort]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()

/** Converts this [ULong] value to [Long]. */

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Converts this [ULong] value to [Int]. */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Converts this [ULong] value to [Short]. */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [ULong] value to [Byte]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [ULong] value to [UInt]. */

@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [ULong] value to [UShort]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()

/** Converts this [ULong] value to [Long]. */

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Converts this [ULong] value to [Int]. */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Converts this [ULong] value to [Short]. */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [ULong] value to [Byte]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [ULong] value to [UInt]. */

@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [ULong] value to [UShort]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()

/** Converts this [ULong] value to [Long]. */

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Converts this [ULong] value to [Int]. */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Converts this [ULong] value to [Short]. */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [ULong] value to [Byte]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [ULong] value to [UInt]. */

@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [ULong] value to [UShort]. */

@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UByte]. */

@kotlin.internal.InlineOnly
public inline fun toRangeTo(other: ULong): ULongRange = ULongRange(this, other)

/** Shifts this value left by the [bitCount] number of bits. */

@kotlin.internal.InlineOnly
public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)

/** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */

@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)

/** Performs a bitwise AND operation between the two values. */

@kotlin.internal.InlineOnly
public inline fun and(other: ULong): ULong = ULong(this.data and other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline function or(other: ULong): ULong = ULong(this.data or other.data)

/** Performs a bitwise XOR operation between the two values. */

@kotlin.internal.InlineOnly
public inline function xor(other: ULong): ULong = ULong(this.data xor other.data)

/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline function or(other: ULong): ULong = this.toULong().or(other.toULong())
/**
 * Converts this [ULong] value to [Double].
 * The resulting value is the closest 'Double' to this 'ULong' value.
 * In case when this 'ULong' value is exactly between two 'Double's, the one with zero at least significant bit of mantissa is selected.
 * @kotlin.internal.InlineOnly
 * public inline fun toDouble(): Double = ulongToDouble(data) // public override fun toString(): String = ulongToString(data)
 */

/**
 * Converts this [Byte] value to [ULong].
 * If this value is positive, the resulting 'ULong' value represents the same numerical value as this 'Byte'.
 * The least significant 8 bits of the resulting 'ULong' value are the same as the bits of this 'Byte' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Short] value to [ULong].
 * The least significant 16 bits of the resulting 'ULong' value are the same as the bits of this 'Short' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Int] value to [ULong].
 * The least significant 32 bits of the resulting 'ULong' value are the same as the bits of this 'Int' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Long] value to [ULong].
 * The resulting 'ULong' value has the same binary representation as this 'Long' value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toULong(): ULong = ULong(this)

/**
 * Converts this [Float] value to [ULong].
 * Returns zero if this 'Float' value is negative or 'NaN', [ULong.MAX_VALUE] if it's bigger than [ULong.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

/**
 * Converts this [Double] value to [ULong].
 * Returns zero if this 'Double' value is negative or 'NaN', [ULong.MAX_VALUE] if it's bigger than [ULong.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toULong(): ULong = doubleToULong(this)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt // See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns 1st *element* from the list.
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 1.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component1(): T {
    return get(0)
}

/**
 * Returns 2nd *element* from the list.
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 2.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component2(): T {
    return get(1)
}

/**
 * Returns 3rd *element* from the list.
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 3.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component3(): T {
    return get(2)
}

/**
 * Returns 4th *element* from the list.
 * Throws an [IndexOutOfBoundsException] if the size of this list is less than 4.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component4(): T {
    return get(3)
}

throw an [IndexOutOfBoundsException] if the size of this list is less than 1.

* @kotlin.internal.InlineOnly
* @package kotlin.collections
* @name it
* @return <T>
* @throws IllegalArgumentException
* @throws IndexOutOfBoundsException
*/

/**
 * Converts this [ULong] value to [Double].
 * The resulting value is the closest 'Double' to this 'ULong' value.
 * In case when this 'ULong' value is exactly between two 'Double's, the one with zero at least significant bit of mantissa is selected.
 * @kotlin.internal.InlineOnly
 * public inline fun toDouble(): Double = ulongToDouble(data)
 */

/**
 * Converts this [Byte] value to [ULong].
 * If this value is positive, the resulting 'ULong' value represents the same numerical value as this 'Byte'.
 * The least significant 8 bits of the resulting 'ULong' value are the same as the bits of this 'Byte' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Short] value to [ULong].
 * The least significant 16 bits of the resulting 'ULong' value are the same as the bits of this 'Short' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Int] value to [ULong].
 * The least significant 32 bits of the resulting 'ULong' value are the same as the bits of this 'Int' value, whereas the most significant 48 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Long] value to [ULong].
 * The resulting 'ULong' value has the same binary representation as this 'Long' value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toULong(): ULong = ULong(this)

/**
 * Converts this [Float] value to [ULong].
 * Returns zero if this 'Float' value is negative or 'NaN', [ULong.MAX_VALUE] if it's bigger than [ULong.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

/**
 * Converts this [Double] value to [ULong].
 * Returns zero if this 'Double' value is negative or 'NaN', [ULong.MAX_VALUE] if it's bigger than [ULong.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toULong(): ULong = doubleToULong(this)
Open Source Used In DNAC Platform Arica 9616

```kotlin
/**
 * Returns index of the first element matching the given [predicate], or -1 if the collection does not contain such element.
 */
public inline fun <T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.
 */
public fun <T> List<T>.getOrDefault(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex)
        get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the list does not contain element.
 */
public fun <T> List<T>.getOrNull(index: Int): T? {
    return if (index >= 0 && index <= lastIndex)
        get(index) else null
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.
 */
public fun <T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex)
        get(index) else defaultValue(index)
}

/**
 * Returns the first element matching the given [predicate], or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.
 */
public fun <T> List<T>.firstOrNull(): T? {
    return if (isEmpty()) null else this[0]
}

/**
 * Returns the first element, or `null` if the list is empty.
 */
public fun <T> List<T>.firstOrNull(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if the list is empty.
 */
public fun <T> List<T>.first(): T? {
    return if (isEmpty()) null else this[0]
}
```

* Returns last index of [element], or -1 if the collection does not contain element.

```kotlin
public fun <T> Iterable<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}
```

```kotlin
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    return lastIndexOf(element)
}
```

```kotlin
/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
public fun <T> List<T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}
```

```kotlin
/**
 * Returns last index of [element], or `null` if the list does not contain element.
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    val iterator = this.listIterator(size)
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (predicate(element)) return element
    }
    return null
}
```

```kotlin
/**
 * Returns a random element from this collection.
 * 
 * @throws NoSuchElementException if this collection is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.random(): T {
    return random() as T
}
```

```kotlin
/**
 * Returns a random element from this collection using the specified source of randomness.
 * 
 * @throws NoSuchElementException if this collection is empty.
 * @throws NoSuchElementException if this collection contains more than one element.
 */
@SinceKotlin("1.3")
public fun <T> Collection<T>.random(random: Random): T {
    if (isEmpty())
        throw NoSuchElementException("Collection is empty.")
    return elementAt(random.nextInt(size))
}
```

```kotlin
/**
 * Returns a single element, or throws an exception if the collection is empty or has more than one element.
 */
public fun <T> List<T>.single(): T {
    return when (size) {
        0 -> throw NoSuchElementException("List is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("List has more than one element.")
    }
}
```

```kotlin
/**
 * Returns a single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Iterable<T>.singleOrNull(): T? {
    when (this) {
        is List -> return this.single()
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                throw NoSuchElementException("Collection contains no element matching the predicate.")
            val single = iterator.next()
            if (iterator.hasNext())
                throw IllegalArgumentException("Collection contains more than one matching element.")
            return single
        }
    }
}
```

```kotlin
/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 * 
 * Returns a random element from this collection, or `null` if it has no such element.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull() as T
}
```

```kotlin
/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 * 
 * Returns a random element from this collection, or `null` if it has no such element.
 */
@SinceKotlin("1.3")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T> Collection<T>.randomOrNull(random: Random): T? {
    if (isEmpty())
        return null
    return elementAt(random.nextInt(size))
}
```

```kotlin
/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var found = false
    for (item in this) {
        if (predicate(item)) {
            if (found) throw IllegalArgumentException("Collection contains more than one matching element.")
            found = true
            return item
        }
    }
    throw NoSuchElementException("Collection contains no element matching the predicate.")
}
```
val iterator = iterator()
if (!iterator.hasNext()) return null
val single = iterator.next()
if (iterator.hasNext()) return null
return single
}

/**
 * Returns single element, or `null` if the list is empty or has more than one element.
 */
public fun <T> List<T>.singleOrNull(): T? {
    return if (size == 1) this[0] else null
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    return if (predicate(null)) null
        else listOf().toInt()
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> List<T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    val list: ArrayList<T>
    if (this is Collection<*>) {
        val resultSize = size - n
        if (resultSize <= 0)
            return emptyList()
        if (resultSize == 1)
            return listOf(last())
        list = ArrayList<T>(resultSize)
        if (this is List<T>) {
            if (this is RandomAccess) {
                for (index in n until size)
                    list.add(this[index])
            } else {
                for (item in listIterator(n))
                    list.add(item)
            }
        }
    }
    else {
        list = ArrayList<T>()
    }
    var count = 0
    for (item in this) {
        if (count >= n) list.add(item) else ++count
    }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Iterable<T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> {
    return filterIndexedTo(ArrayList<T>(), predicate)
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T, C : MutableCollection<in T>> Iterable<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 */
public inline fun <reified R> Iterable<T>.filterIsInstance(): List<R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Appends all elements that are instances of specified type parameter R to the given [destination].
 */
public inline fun <reified R, C : MutableCollection<in T>> Iterable<T>.filterIsInstanceTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements except first [n] elements that satisfy the given [predicate].
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> List<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    return if (predicate(null)) null
        else listOf().toInt()
}

/**
 * Returns a list containing only elements matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    return if (predicate(null)) null
        else listOf().toInt()
}

/**
 * Returns a list containing all elements except first [n] elements that satisfy the given [predicate].
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> List<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    return if (predicate(null)) null
        else listOf().toInt()
}

/**
 * Returns a list containing all elements except first [n] elements that satisfy the given [predicate].
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> List<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    return if (predicate(null)) null
        else listOf().toInt()
}

/**
 * Returns a list containing all elements except first [n] elements that satisfy the given [predicate].
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> List<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
Open Source Used In DNAC Platform Arica  9619
Collection<Int>.toIntArray(): IntArray {
    val result = IntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Long containing all of the elements of this collection.
 */
public fun Collection<Long>.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns a [Map] containing key-value pairs provided by [transform] function
 */
public fun Collection<in T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element of the given collection.
 */
public inline fun <T, K, M : MutableMap<in K, in T>> Iterable<T>.associateTo(destination: M, keySelector: (T) -> K): M {
    for (element in this)
        destination.put(keySelector(element), element)
    return destination
}

/**
 * Returns a [Map] containing the elements from the given collection indexed by the key
 */
public fun Collection<in T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
}

/**
 * Returns a [Map] containing the values provided by [valueSelector] function applied to elements of the given collection.
 */
public fun Collection<in T>.associateByTo(destination: M, valueSelector: (T) -> V): M {
    for (element in this)
        destination.put(valueSelector(element), element)
    return destination
}
samples.collections.Collections.Transformations.associateWith

public inline fun <K, V> Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * A very simple implementation of the [associateWith] function.
 * 
 * @since Kotlin("1.3")
 * @sample samples.collections.Collections.Transformations.associateWith
 */
public inline fun <K, V, M : MutableMap<in K, in V>>
    Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <T, C : MutableCollection<in T>>
    Iterable<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a [HashSet] of all elements.
 */
public fun <T> Iterable<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.
 */
public inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original collection.
 */
public inline fun <T, R> Iterable<T>.flatMapIndexed(transform: (index: Int, T) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}
[transform] function being invoked on each element\n* and its index in the original collection, to the given
[destination].\n
*/\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedIterableTo")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination\n}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element\n* and its index in the original collection, to the given [destination].\n
 */\n@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapIndexedSequenceTo")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapIndexedTo(destination: C, transform: (index: Int, T) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(checkIndexOverflow(index++), element)
        destination.addAll(list)
    }
    return destination\n}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element of original collection, to the given [destination].\n
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapTo")\n@kotlin.internal.InlineOnly\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination\n}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each
 * element of original collection, to the given [destination].\n
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@kotlin.jvm.JvmName("flatMapSequenceTo")\npublic inline fun <T, R, C : MutableCollection<in R>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination\n}

/**
 * Groups elements of the original collection by the key returned by the given [keySelector]
 * function\n* applied to each element and returns a map where each group key is associated with a list of
 * corresponding elements\n* \n* \n* The returned map preserves the entry iteration order of the keys produced from the
 * original collection.\n* \n* \n* @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K> Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)\n}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection\n* by the key returned by the given [keySelector] function applied to the element\n* and returns a map where each group key is associated with a list of corresponding values.\n* \n* \n* The returned map preserves the entry iteration order of the keys produced from the
 * original collection.\n* \n* \n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<V, MutableList<V>>(), keySelector, valueTransform)\n}

// Sample code

val keySelector = { element: T -> element }\nval valueTransform = { element: T -> element.value }\nval groupMap = groupByTo(groupList)\n
valueTransform: (T -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<in V>() }\n        list.add(valueTransform(element))\n    }\n    return destination\n} \n
* Creates a [Grouping] source from a collection to be used later with one of group-and-fold operations\n* using the specified [keySelector] function to extract a key from each element.\n
```kotlin
Iterable<T>.groupingBy(keySelector: (T) -> K): Grouping<T, K> {\n    override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()\n    override fun keyOf(element: T): K = keySelector(element)\n    return destination
}
```
samples.collections.Collections.Transformations.distinctAndDistinctBy

/*\npublic fun <T> Iterable<T>.distinct():
List<T> {
    return this.toMutableSet().toList()
}

/**
 * Returns a set containing all elements that are contained by both this collection and the specified collection.

 * @param other
 * @return
 */
public infix fun <T> Iterable<T>.intersect(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a set containing all elements that are contained by this collection and not contained by the specified collection.

 * @param other
 * @return
 */
public infix fun <T> Iterable<T>.subtract(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

/**
 * Returns `true` if all elements match the given predicate.

 * @param predicate
 * @return
 */
public inline fun <T> Iterable<T>.all(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return true
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if collection has at least one element.

 * @return
 */
public fun <T> Iterable<T>.any(): Boolean {
    if (this is Collection) return !isEmpty()
    return iterator().hasNext()
}

/**
 * Returns `true` if at least one element matches the given predicate.

 * @param predicate
 * @return
 */
public inline fun <T> Iterable<T>.any(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements in this collection.

 */
public fun <T> Iterable<T>.count(): Int {
    if (this is Collection) return size
    var count = 0
    for (element in this) checkCountOverflow(++count)
    return count
}

/**
 * Returns the number of elements matching the given predicate.

 * @param predicate
 * @return
 */
public inline fun <T> Iterable<T>.count(predicate: (T) -> Boolean): Int {
    if (this is Collection && isEmpty()) return 0
    var count = 0
    for (element in this) if (predicate(element)) checkCountOverflow(++count)
    return count
}

/**
 * Accumulates value starting with initial value and applying operation from left to right.

 * @param operation
 * @return
 */
public inline fun <T, R> Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with initial value and applying operation from left to right.

 * @param operation
 * @return
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.count(): Int {
    return size
}

/**
 * Returns the number of elements in this collection.

 */
@kotlin.internal.InlineOnly
public inline fun <T, R> Collection<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
*/
public inline fun <T, R> Iterable<T>.foldIndexed(initial: R, operation: (Int, acc: R, T) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(0, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 * Returns the specified [initial] value if the list is empty.
 * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <T, R> List<T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var accumulator = initial
    if (isEmpty()) return initial
    val iterator = listIterator(size)
    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original list and current accumulator value.
 * Returns the specified [initial] value if the list is empty.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <T, R> List<T>.foldRightIndexed(initial: R, operation: (Int, T, acc: R) -> R): R {
    var accumulator = initial
    if (isEmpty()) return initial
    val iterator = listIterator(size)
    while (iterator.hasNext()) {
        val index = iterator.previousIndex()
        accumulator = operation(index, iterator.next(), accumulator)
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
@kotlin.internal.HidesMembers
public inline fun <T> Iterable<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential
 * index with the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
public inline fun <T> Iterable<T>.forEachIndexed(action: (Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Double>.max(): Double? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Float>.max(): Float? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Iterable<T>.max(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    if (!iterator.hasNext()) return maxElem
    var maxValue = selector(maxElem)
    do {
        val e = iterator.next()
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}

/**
 * Returns the largest value among all values produced by
 * [selector] function.
 * @param [selector] function that produces values.
 * @param [captured] object that is captured by the lambda expression.
 * @throws NoSuchElementException if the list is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    do {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxValue
if the collection is empty.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOf(selector: (T) -> Float): Float {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Iterable<T>.maxOf(selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.maxOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Iterable<T>.maxOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = comparator.compare(maxValue, v) < 0 {
            maxValue = v
        }
    }
    return maxValue
}
```

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Iterable<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator =
iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        var max = iterator.next() var max = maxOf(max, e)\n    }
    return maxValue
}

/**
 * Returns the largest element or `null` if there are no elements.
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Iterable<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    if (!iterator.hasNext()) return maxElem
    val maxValue = selector(maxElem)
    do {
        val e = iterator.next()
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the collection.
 * @throws NoSuchElementException if the collection is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        var minValue = minOf(minValue, v)
    }
    return minValue
}

@Deprecated("Use maxWithOrNull instead.", ReplaceWith("maxWithNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Iterable<T>.maxWith(comparator: Comparator<in T>): T? {
    return maxWithOrNull(comparator)
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Double>.min(): Double? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun Iterable<Float>.min(): Float? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T : Comparable<T>> Iterable<T>.min(): T? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <T, R : Comparable<R>> Iterable<T>.minByOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    do {
        val e = iterator.next()
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    } while (iterator.hasNext())
    return minElem
}
/* functor */

fun <T> Iterable<T>.minOf(selector: (T) -> Float): Float {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the collection or 'null' if there are no elements.
 */
fun <T> Iterable<T>.minOf(selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements.
 */
fun <T> Iterable<T>.minOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements.
 */
fun <T, R : Comparable<R>> Iterable<T>.minOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function * applied to each element in the collection.
 */
fun <T, R> Iterable<T>.minOfWith(comparator: Comparator<R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function * applied to each element in the collection or `null` if there are no elements.
 */
fun <T, R> Iterable<T>.minOfWithOrNull(comparator: Comparator<R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
minValue = \( v \) \n\n...
accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original collection.
 * @param operation\n * bool operation\n * \n * @param [operation] function that takes the index of an element, current accumulator value and the element itself,\n * and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.reduceOrNull\n * @param samples.Collections.Aggregates.reduceRightOrNull\n */
public inline fun <S, T : S> Iterable<T>.reduceIndexedOrNull(operation: (index: Int, acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element with its index in the original list and current accumulator value.
 * @param operation\n * int operation\n * \n * @param [operation] function that takes the index of an element, the element itself and current accumulator value,\n * and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.reduceRightIndexedOrNull\n * @param samples.Collections.Aggregates.reduceRightOrNull\n */
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        return null
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        val index = iterator.previousIndex()
        accumulator = operation(index, iterator.previous(), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * @param operation\n * int operation\n * \n * @param [operation] function that takes an element and current accumulator value,\n * and calculates the next accumulator value.
 * @param samples.collections.Collections.Aggregates.reduceRightOrNull\n */
public inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        throw UnsupportedOperationException("Empty list can't be reduced.")
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        accumulator = operation(iterator.previous(), accumulator)
    }
    return accumulator
}
samples.collections.Collections.Aggregates.reduceRightOrNull

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class) val
public inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious()) return null
    var accumulator: S = iterator.previous()
    while (iterator.hasNext()) {
        accumulator = operation(iterator.next(), accumulator)
    }
    return accumulator
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right *to each element and current accumulator value that starts with [initial] value.*

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
public inline fun <S, T : S> List<T>.runningFold(initial: S, operation: (acc: S, T) -> S): List<S> {
    val estimatedSize = collectionSizeOrDefault(9)
    if (estimatedSize == 0) return listOf(initial)
    val result = ArrayList<S>(estimatedSize + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
public inline fun <S, T : S> List<T>.runningReduce(operation: (acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
public inline fun <S, T : S> List<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    var index = 1
    while (iterator.hasNext()) {
        accumulator = operation(index++, accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    var index = 1
    while (iterator.hasNext()) {
        accumulator = operation(index++, accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}
```

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element in the collection and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <T, R> Iterable<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    return runningFoldIndexed(initial, operation)
}

@Deprecated("Use runningReduce instead.", ReplaceWith("runningReduce(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
    return runningReduce(operation)
}

@Deprecated("Use runningReduceIndexed instead.", ReplaceWith("runningReduceIndexed(operation)"), level = DeprecationLevel.ERROR)
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
public inline fun <T> Iterable<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
public inline fun <T> Iterable<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfDouble")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfInt")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfLong")
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] applied to each element in the collection.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("sumOfUInt")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```
sum of all values produced by [selector] function applied to each element in the collection.

@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class) @OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfULong") @ExperimentalUnsignedTypes @kotlin.inter
nal.InlineOnly

public inline fun <T> Iterable<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentExce
ption] if there are any `null` elements.
 */
@SinceKotlin("1.4") @OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType @kotlin.jvm.JvmName("sumOfULong") @ExperimentalUnsignedTypes
public inline fun <T> Iterable<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentExce
ption] if there are any `null` elements.
 */
@SinceKotlin("1.2")
public fun <T : Any> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this collection into several lists each not exceeding the given [size] and applies the given [transfo
rm] function to an each.
 * @return list of results of the [transform] applied to an each list.
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last list may have less elements than the given [size]. The last list in the resulting list may have less elements than the number of elements in this collection.
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T : Any> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Splits this collection into a list of lists each not exceeding the given [size] and the last list in the result
ning list may have less elements than the given [size]. The last list in the resulting list may have less elements than the number of elements in this collection.
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * @return list of lists each not exceeding the given [size].
 */
@SinceKotlin("1.2")
public fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Returns a list containing all elements of the original collection without the first occurrence of the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minus(element: T): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    var removed = false
    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.
 * @param elements the elements to remove from the original collection.
 * @return list of elements of the original collection except the elements contained in the given [elements] array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minus(elements: Array<out T>): List<T> {
    if (elements.isEmpty()) return this.toList()
    val other = elements.toHashSet()
    return this.filterNot { it in other }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] collection.
 * @param elements the elements to remove from the original collection.
 * @return list of elements of the original collection except the elements contained in the given [elements] collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minus(elements: Iterable<T>): List<T> {
    val other = elements.convertToSetForSetOperationWith(this) if (other.isEmpty()) return this.filterNot { it in other }
    return this.filterNot { it in other }
}

/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] sequence.
 * @param elements the elements to remove from the original collection.
 * @return list of elements of the original collection except the elements contained in the given [elements] sequence.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minus(elements: Sequence<T>): List<T> {
    val other = elements.toHashSet()
    return this.filterNot { it in other }
}

/**
 * Returns a list containing all elements of the original collection without the first occurrence of the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minusElement(element: T): List<T> {
    return this.filterNot { it == element }
}
return minus(element)\n\n/* Splits the original collection into pair of lists.\n * where *first* list contains elements for which [predicate] yielded `true`.\n * while *second* list contains elements for which [predicate] yielded `false`.\n */\n\nIter<\T>.partition(predicate: (\T) -> Boolean): Pair<List<\T>, List<\T>> \n\nval first = ArrayList<\T>()\nval second = ArrayList<\T>()\nfor (element in this) \n\nif (predicate(element)) \n\n{\n    first.add(element)\n}\n\nelse \n\n{\n    second.add(element)\n}\n\n\nreturn Pair(first, second)\n\n/* Returns a list containing all elements of the original collection and then the given [element].\n */\n\nIterable<\T>.plus(element: \T): List<\T> \n\n\nval result = ArrayList<\T>()\nresult.addAll(this)\nresult.addAll(element)\n\nreturn result\n\n/* Returns a list containing all elements of the original collection and then all elements of the given [elements] array.\n */\n\nIterable<\T>.plus(elements: Array<out \T>): List<\T> \n\n\nval result = ArrayList<\T>()\nresult.addAll(this)\nresult.addAll(elements)\n\nreturn result\n\n/* Returns a list containing all elements of the original collection and then all elements of the given [elements] collection.\n */\n\nIterable<\T>.plus(elements: Iterable<\T>): List<\T> \n\n\nval result = ArrayList<\T>()\nresult.addAll(this)\nresult.addAll(elements)\n\nreturn result\n\n/* Returns a list containing all elements of the original collection and then all elements of the given [elements] sequence.\n */\n\nIterable<\T>.plus(elements: Sequence<\T>): List<\T> \n\n\nval result = ArrayList<\T>()\nresult.addAll(this)\nresult.addAll(elements)\n\nreturn result\n\n/* Returns a list containing all elements of the original collection and then all elements of the given [elements] stream.\n */\n\nIterable<\T>.plus(elements: Stream<\T>): List<\T> \n\n\nval result = ArrayList<\T>()\nresult.addAll(this)\nresult.addAll(elements)\n\nreturn result
result.add(List(windowSize) { this[it + index] })
index += step

val result = ArrayList<List<T>>(windowSize)
for (i in 0 until windowSize) {
    result.add(windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach { it }).
    return result
}

result = ArrayList<List<T>>()
windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
    result.add(it)
}
return result

/**
 * Returns a list of results of applying the given [transform] function to each list representing a view over the window of the given [size].
 * [partialWindows] sliding along this collection with the given [step].
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * Several last lists may have less elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step,
 * by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any,
 * by default `false` which means partial windows won't be preserved
 * @sample samples.collections.Sequences.Transformations.averageWindows */
@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<R>(resultCapacity)
        val window = MovingSubList(this)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (!partialWindows && windowSize < size) break
            window.move(index, index + windowSize)
    result.add(transform(window))
    index += step
}
return result

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] collection with the same index.
 * The returned list has length of the shortest collection.
 * Returns a list of pairs built from the elements of `this` collection and the [other] collection with the same index.
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 * @sample samples.collections.Iterables.Operations.zipIterable
 */
public infix fun <T, R> Iterable<T>.zip(other: Iterable<R>): List<Pair<T, R>> {
    val first = iterator()
    val second = other.iterator()
    val list = ArrayList<Pair<T, R>>(minOf(this.collectionSizeOrDefault(10), other.collectionSizeOrDefault(10)))
    while (first.hasNext() && second.hasNext()) {
        list.add(transform(first.next(), second.next()))
    }
    return list
}

/**
 * Returns a list of pairs of each two adjacent elements in this collection.
 * The returned list is empty if this collection contains less than two elements.
 * @sample samples.collections.Collections.Transformations.zipWithNext
 */
@SinceKotlin("1.2")
public fun <T> Iterable<T>.zipWithNext(): List<Pair<T, T>> {
    return zipWithNext { a, b -> a to b }
}

/**
 * Returns a list containing the results of applying the given [transform] function to each pair of two adjacent elements in this collection.
 * The returned list is empty if this collection contains less than two elements.
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun <T> Iterable<T>.zipWithNextToFindDeltas(): List<T>

inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> R): List<R> {
    val iterator = iterator()
    if (!iterator.hasNext()) return emptyList()
    val result = mutableListOf<R>()
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result
}

/**
 * Appends the string from all the elements separated using [separator] and
 * using the given [prefix] and [postfix] if supplied.
 * 
 * If the collection could be huge, you can specify a non-negative value of [limit],
 * in which case only the first [limit] elements will be appended, followed by the
 * [truncated] string (which defaults to "...").
 */
public fun <T, A : Appendable>
Iterable<T>.joinTo(buffer: A, separator: CharSequence = " ", prefix: CharSequence = "", postfix: CharSequence = " ", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) {
            buffer.appendElement(element, transform)
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Returns this collection as an [Iterable].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.asIterable(): Iterable<T> {
    return this
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if
 * supplied.
 * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun <T> Iterable<T>.asSequence(): Sequence<T> {
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Iterable<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}
public fun Iterable<Short>.sum(): Int {
    var sum = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Long>.sum(): Long {
    var sum = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Float>.sum(): Float {
    var sum = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Double>.sum(): Double {
    var sum = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.comparisons.naturalOrder
import kotlin.random.Random

/** Returns the array if it's not `null`, or an empty array otherwise. */
@kotlin.internal.InlineOnly
public actual inline fun <T> Array<out T>?.orEmpty(): Array<out T> = this ?: emptyArray<T>()

@kotlin.internal.InlineOnly
public actual inline fun <T> Collection<T>.toTypedArray(): Array<T> = copyToArray(this)

@JsName("copyToArray")
@PublishedApi
internal fun <T> copyToArray(collection: Collection<T>): Array<T> {
    return if (collection.asDynamic().toArray !== undefined)
        collection.asDynamic().toArray().unsafeCast<Array<T>>()
    else
        copyToArrayImpl(collection).unsafeCast<Array<T>>()
}

@JsName("copyToArrayImpl")
internal actual fun <T> copyToArrayImpl(collection: Collection<*>): Array<Any?> {
    val array = emptyArray<Any?>()
    val iterator = collection.iterator()
    while (iterator.hasNext())
        array.asDynamic().push(iterator.next())
    return array
}

@JsName("copyToExistingArrayImpl")
internal actual fun <T> copyToArrayImpl(collection: Collection<*>, array: Array<T>): Array<T> {
    if (array.size < collection.size)
        return copyToArrayImpl(collection).unsafeCast<Array<T>>()

    val iterator = collection.iterator()
    var index = 0
    while (iterator.hasNext()) {
        array[index++].unsafeCast<T>()
    }
    if (index < array.size) {
        array[index].null.unsafeCast<T>()
    }
    return array
}

/**
 * Returns an immutable list containing only the specified object [element].
 */
public fun <T> listOf(element: T): List<T> = arrayListOf(element)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildListInternal(builderAction: MutableList<E>.() -> Unit): List<E> {
    return ArrayList<E>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildListInternal(capacity: Int, builderAction: MutableList<E>.() -> Unit): List<E> {
    val list = ArrayList<E>(capacity)
    list().apply(builderAction).build()
    return list
}

/**
 * Returns an immutable set containing only the specified object [element].
 */
public fun <T> setOf(element: T): Set<T> = hashSetOf(element)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildSetInternal(builderAction: MutableSet<E>.() -> Unit): Set<E> {
    return LinkedHashSet<E>().apply(builderAction).build()
}

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <E> buildSetInternal(capacity: Int, builderAction: MutableSet<E>.() -> Unit): Set<E> {
    val set = LinkedHashSet<E>(capacity)
    set().apply(builderAction).build()
    return set
}

/**
 * Returns an immutable map, mapping the specified key to the specified value.
 */
public fun <K, V> mapOf(pair: Pair<K, V>): Map<K, V> = hashMapOf(pair)

@PublishedApi
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
internal actual inline fun <K, V> buildMapInternal(builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    val map = LinkedHashMap<K, V>()
    map().apply(builderAction).build()
    return map
}
fun <T> Iterator<T>.iterator(): Iterator<T> = this

fun <T> Iterator<T>.withIndex(): IndexingIterator<T>

fun <T> Iterator<T>.forEach(operation: (T) -> Unit): Unit {
    for (element in this) operation(element)
}

internal class IndexingIterator<out T>(private val iterator: Iterator<T>) : Iterator<IndexedValue<T>> {
    private var index = 0
    final override fun hasNext(): Boolean = iterator.hasNext()
    final override fun next(): IndexedValue<T> = IndexedValue(checkIndexOverflow(index++), iterator.next())
}

expect fun <T : Comparable<T>> maxOf(a: T, b: T): T

expect inline fun maxOf(a: Byte, b: Byte): Byte

expect inline fun maxOf(a: Short, b: Short): Short

expect inline fun maxOf(a: Int, b: Int): Int

expect inline fun maxOf(a: Long, b: Long): Long

expect inline fun maxOf(a: Float, b: Float): Float

expect inline fun maxOf(a: Double, b: Double): Double

expect <T> maxOf(a: T, maxOf(b, c, comparator), comparator)

expect <T> maxOf(a: T, comparator<in T>, T)

expect fun <T> maxOf(a: T, Comparator<in T>): T

expect fun <T> maxOf(a: T, vararg other: T): T

expect fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T

expect fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T

```kotlin
@SinceKotlin("1.4")
public expect fun maxOf(a: Byte, vararg other: Byte): Byte

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public expect fun maxOf(a: Short, vararg other: Short):
    Short

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public expect fun maxOf(a: Int, vararg other: Int):
    Int

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public expect fun maxOf(a: Long, vararg other: Long):
    Long

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public expect fun maxOf(a: Float, vararg other: Float):
    Float

/**
 * Returns the greater of the given values.
 */
@SinceKotlin("1.4")
public expect fun maxOf(a: Double, vararg other: Double):
    Double

/**
 * Returns the greater of the given values according to the order specified by the given [comparator].
 */
@SinceKotlin("1.4")
public fun <T> maxOf(a: T, vararg other: T, comparator: Comparator<in T>):
    T

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Byte, b: Byte):
    Byte

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Short, b: Short):
    Short

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Int, b: Int):
    Int

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Long, b: Long):
    Long

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Float, b: Float):
    Float

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Double, b: Double):
    Double

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Byte, b: Byte, c: Byte):
    Byte

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Short, b: Short, c: Short):
    Short

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Int, b: Int, c: Int):
    Int

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Long, b: Long, c: Long):
    Long

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Float, b: Float, c: Float):
    Float

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: Double, b: Double, c: Double):
    Double

/**
 * Returns the smaller of three values according to the order specified by the given [comparator].
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Byte, vararg other: Byte):
    Byte

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Short, vararg other: Short):
    Short

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Int, vararg other: Int):
    Int

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Long, vararg other: Long):
    Long

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Float, vararg other: Float):
    Float

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Double, vararg other: Double):
    Double

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Float, vararg other: Float):
    Float

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Double, vararg other: Double):
    Double
```

Open Source Used In DNAC Platform Arica 9641
Int, vararg other: Int): Int

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Long, vararg other: Long): Long

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Float, vararg other: Float): Float

/**
 * Returns the smaller of the given values.
 */
@SinceKotlin("1.4")
public expect fun minOf(a: Double, vararg other: Double): Double

/**
 * Returns the smaller of the given values according to the order specified by the given [comparator].
 */
@SinceKotlin("1.4")
public fun <T> minOf(a: T, vararg other: Comparator<T>): T {
    var min = a
    for (e in other) if (comparator.compare(min, e) > 0) min = e
    return min
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
public fun <K, V> Map<K, V>.toList(): List<Pair<K, V>> {
    if (size == 0)
        return emptyList()
    val iterator = entries.iterator()
    if (!iterator.hasNext())
        return emptyList()
    val first = iterator.next()
    if (!iterator.hasNext())
        return listOf(first.toPair())
    val result = ArrayList<Pair<K, V>>(size)
    result.add(first.toPair())
    do {
        result.add(iterator.next().toPair())
    } while (iterator.hasNext())
    return result
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <K, V, R> Map<K, V>.flatMap(transform: (Map.Entry<K, V>) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequenceTo")
public inline fun <K, V, R, C : MutableCollection<in R>> Map<K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("map")
public inline fun <K, V, R> Map<K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("mapNotNull")
public inline fun <K, V, R : Any> Map<K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

"/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
public fun <K, V> Map<K, V>.toList(): List<Pair<K, V>> {
    if (size == 0)
        return emptyList()
    val iterator = entries.iterator()
    if (!iterator.hasNext())
        return emptyList()
    val first = iterator.next()
    if (!iterator.hasNext())
        return listOf(first.toPair())
    val result = ArrayList<Pair<K, V>>(size)
    result.add(first.toPair())
    do {
        result.add(iterator.next().toPair())
    } while (iterator.hasNext())
    return result
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequence")
public inline fun <K, V, R> Map<K, V>.flatMap(transform: (Map.Entry<K, V>) -> Sequence<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each entry of original map.
 * If there are multiple equal minimal values, returns the first of them.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapSequenceTo")
public inline fun <K, V, R, C : MutableCollection<in R>> Map<K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("map")
public inline fun <K, V, R> Map<K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function on each entry of original map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("mapNotNull")
public inline fun <K, V, R : Any> Map<K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}
R>> Map<out K, V>.mapNotNullTo(destination: C, transform: (Map.Entry<K, V>) -> R?): C {
    for (item in this) {
        return destination
    }
    return null
}

/**
 * Applies the given [transform] function to each entry of the original map and appends the resulting values to the given [destination].
 */
public inline fun <K, V, R, C : MutableCollection<R>> Map<out K, V>.mapTo(destination: C, transform: (Map.Entry<K, V>) -> R): C {
    forEach {
        transform(it) ?: let { destination.add(it) } } return destination
}

/**
 * Returns `true` if all entries match the given [predicate].
 */
public fun <K, V> Map<out K, V>.all(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if map has at least one entry.
 */
public fun <K, V> Map<out K, V>.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if at least one entry matches the given [predicate].
 */
public inline fun <K, V> Map<out K, V>.any(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of entries in this map.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.count(): Int {
    return size
}

/**
 * Returns the number of entries matching the given [predicate].
 */
public inline fun <K, V> Map<out K, V>.count(predicate: (Map.Entry<K, V>) -> Boolean): Int {
    if (isEmpty()) return 0
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Performs the given [action] on each entry.
 */
@kotlin.internal.HidesMembers
public inline fun <K, V> Map<out K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {
    for (element in this) action(element)
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin("1.4")
@kotlin.internal.InlineOnly
    return maxByOrNull(selector)
}

/**
 * Returns the first entry yielding the largest value of the given function or `null` if there are no entries.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Double): Double {
    return entries.maxOf(selector)
}

/**
 * Performs the given [action] on each entry in the map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.forEachWithPredicate(action: (Map.Entry<K, V>) -> R?): R {
    for (element in this) if (action(element)) return element
    return null
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each entry in the map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Float): Float {
    return entries.maxOf(selector)
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each entry in the map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> R): R {
    return entries.maxOf(selector)
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each entry in the map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Double): Double {
    return entries.maxOf(selector)
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each entry in the map.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.maxOf(selector: (Map.Entry<K, V>) -> Float): Float {
    return entries.maxOf(selector)
}
V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? {
    return entries.maxOfOrNull(selector)??null
}

/**
 * Returns the largest value among all values produced by [selector] 
 * functionn* applied to each entry in the map or `null` if there are no entries.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V> Map<k out K, 
V>.maxOfOrNull(selector: (Map.Entry<K, V>) -> Float): Float? {
    return entries.maxOfOrNull(selector)??null
}

/**
 * Returns the largest value among all values produced by [selector] 
 * functionn* applied to each entry in the map or `null` if there are no entries.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V, R : Comparable<R>> Map<k out K, 
V>.maxOfWith(comparator: Comparator<R>, selector: (Map.Entry<K, V>) -> R): R? {
    return entries.maxOfWith(comparator, selector)??null
}

/**
 * Returns the largest value according to the provided [comparator]n* among all values produced by [selector] function applied to each entry in the map.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V, R : Comparable<R>> Map<k out K, 
V>.maxOfWithOrNull(comparator: Comparator<R>, selector: (Map.Entry<K, V>) -> R): R? {
    return entries.maxOfWithOrNull(comparator, selector)??null
}

@Deprecated("Use maxWithOrNull instead.", 
ReplaceWith("maxWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V> Map<k out K, 
    return maxWithOrNull(comparator)
}

/**
 * Returns the smallest value according to the provided [comparator] or `null` if there are no entries.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V, R : Comparable<R>> Map<k out K, 
    return minByOrNull(selector)
}

/**
 * Returns the smallest value among all values produced by [selector] functionn* applied to each entry in the map.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution 
ByLambdaReturnType@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly fun <K, V> Map<k out K, 
V>.minOf(selector: (Map.Entry<K, V>) -> Double): Double? {
    return entries.minOf(selector)??null
}
(Map.Entry<K, V>) -> Float): Float { return entries.minOf(selector) * Returns the smallest value among all values produced by [selector] function applied to each entry in the map.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V, R : Comparable<R>> Map<out K, V>.minOf(selector: (Map.Entry<K, V>) -> R): R? * If any of values produced by [selector] function is 'NaN', the returned result is 'NaN'.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> Double): Double? * Returns the smallest value among all values produced by [selector] function applied to each entry in the map or 'null' if there are no entries.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V, R : Comparable<R>> Map<out K, V>.minOfOrNull(selector: (Map.Entry<K, V>) -> R): R? * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each entry in the map or 'null' if there are no entries.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V, R> Map<out K, V>.minOfWith(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): R? * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each entry in the map or 'null' if there are no entries.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V, R> Map<out K, V>.minOfWithOrNull(comparator: Comparator<in R>, selector: (Map.Entry<K, V>) -> R): Map.Entry<K, V>? * Returns the first entry having the smallest value according to the provided [comparator] or 'null' if there are no entries.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V> Map<out K, V>.none(): Boolean * Returns `true` if the map has no entries.

@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly@kotlin.experimental.InlineFun fun <K, V> Map<out K, V>.none(predicate: (Map.Entry<K, V>) -> Boolean): Boolean * Performs the given [action] on each entry and returns the map itself afterwards.
```kotlin
/**
 * Performs the given [action] on each entry, providing sequential index with the entry.
 * @param [action] function that takes the index of an entry and the entry itself
 * and performs the action on the entry.
 * @return [Map] or [List] wrapping the original map returning its entries when being iterated.
 */
@SinceKotlin("1.4")
public inline fun <K, V, M : Map<out K, V>>
M.onEachIndexed(action: (index: Int, Map.Entry<K, V>) -> Unit): M {
    return apply {
        entries.forEachIndexed(action)
    }
}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.asIterable(): Iterable<Map.Entry<K, V>> {
    return entries
}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries when being iterated.
 */
public fun <K, V> Map<out K, V>.asSequence(): Sequence<Map.Entry<K, V>> {
    return entries.asSequence()
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

package kotlin.sequences

/*
 * NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt// See:
 * https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
 */

import kotlin.random.*

/**
 * Returns `true` if [element] is found in the sequence.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public operator fun <T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this sequence.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public fun <T> Sequence<T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("Sequence doesn't contain element at index $index.") }
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this sequence.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public fun <T> Sequence<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    if (index < 0)
        return defaultValue(index)
    val iterator = iterator()
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return defaultValue(index)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public fun <T> Sequence<T>.find(predicate: (T) -> Boolean): T? {
    return findLast(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.find(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Sequence contains no element matching the predicate.")
}

/**
 * Returns the first element, or `null` if the sequence is empty.
 */
public fun <T> Sequence<T>.firstOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    return iterator.next()
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.first(predicate: (T) -> Boolean): T? {
    return firstOrNull().find { predicate(it) }
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.firstOrDefault(predicate: (T) -> Boolean): T? {
    return firstOrNull().find { predicate(it) }
}
```

Open Source Used In DNAC Platform Arica  9646
Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        } else if (found) return null
    }
    return single
}

/**
 * Returns a sequence containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @param predicate function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public fun <T> Sequence<T>.drop(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return when {
        n == 0 -> this
        this is DropTakeSequence -> this.drop(n)
        else -> DropSequence(this, n)
    }
}

/**
 * Returns a sequence containing all elements except first elements that satisfy the given [predicate].
 * @param predicate function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public fun <T> Sequence<T>.filterIndexedTo(destination: Collection<T>, predicate: (index: Int, T) -> Boolean): Collection<T> {
    forEachIndexed {
        index, element ->
            if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a sequence containing only elements matching the given [predicate].
 * The operation is intermediate and stateless.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <R, C : Collection<R>> Sequence<R>.filterIsInstanceTo(destination: Collection<R>): Collection<R> {
    for (element in this) if (element is R) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing all elements not matching the given [predicate].
 * @param predicate function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, false, predicate)
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * The operation is intermediate and stateless.
 * @param predicate function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public fun <R, C : Collection<R>> Sequence<R>.filterNotNullTo(destination: Collection<R>): Collection<R> {
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * The operation is intermediate and stateless.
 */
public fun <T, C : Collection<T>> Sequence<T>.filterTo(destination: Collection<T>, predicate: (T) -> Boolean): Collection<T> {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}
Sequence<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element in this) if (!predicate(element)) destination.add(element)\n    return destination\n} \n
The operation is _terminal_ \n* \n* \n* @sample \n* \n samples.collections.Collections.Filtering.filterTo\n* \n\n
Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {\n    for (element in this) if (predicate(element)) destination.add(element)\n    return destination\n} \n
The operation is _intermediate_ and _stateful_ \n* \n* \n* @sample \n samples.collections.Collections.Tranformations.take\n* \n\n
/**
 * Appends all elements matching the given [predicate] to the given [destination]. \n* \nThe operation is _terminal_ \n* \n* @sample \n samples.collections.Collections\n* \n\n**
 * Returns a sequence containing first \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence containing first elements satisfying the given [predicate]. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order of the value returned by specified [selector] function. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator]. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator]. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [transform] function. \n* \nThe operation is _intermediate_ and _stateful_ \n* \n* @sample \n samples.collections.Collections\n* \n\n**
MutableList<T> {
    return toCollection(ArrayList<T>())
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original sequence.
 * The operation is _terminal_ and _stateless_.
 */
public fun <T> Sequence<T>.toSet(): Set<T> {
    return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()
}

/**
 * Returns a single sequence of all elements from results of [transform] function being invoked on each element of original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIterable")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in the original sequence, to the given [destination].
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexedTo")
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedTo(destination: C, transform: (T, Int) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element in the original sequence, to the given [destination].
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@kotlin.jvm.JvmName("flatMapIndexedSequenceTo")
@kotlin.internal.InlineOnly
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapIndexedSequenceTo(destination: C, transform: (T, Int) -> Sequence<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return
}
destination

* Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [destination].

The operation is _terminal_.

Clear inline fun <T, R, C> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original sequence, to the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The operation is _terminal_.
 * The returned map preserves the entry iteration order of the keys produced from the original sequence.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, C : MutableCollection<R>> Sequence<T>.groupBy(keySelector: (T) -> K, C): Map<K, List<T>> {
    return groupByTo(destination=LinkedHashMap<K, MutableList<T>>(), keySelector, valueTransform)
}

/**
 * Groups elements of the original sequence by the key returned by the given [keySelector] function
 * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <T, K, M : MutableMap<K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold
 * operations
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Grouping.groupingByEachCount
 */
@SinceKotlin("1.1")
public inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to each element in the original sequence.
 * @sample samples.collections.Collections.Transformations.map
 */
public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {
    return TransformingSequence(this, transform)
}

/**
 * Returns a sequence containing
 * the results of applying the given [transform] function
 * to each element and its index in the original sequence.
 * @param [transform] function that takes the index of an element and the element itself
 * and returns the result of applying the transform to the element.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Transformations.mapIndexed
 */
public fun <T, R> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R): Sequence<R> {
    return TransformingIndexedSequence(this, transform)
}
@param [transform] function that takes the index of an element and the element itself
* and returns the result of the transform applied to the element.
* The operation is _intermediate_ and _stateless_.

**public fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {**

    var sequence = doubleBufferedTransformingIndexedSequence镛(target: R)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element and its index in the original sequence
    * and appends only the non-null results to the given destination.
    * The operation is _terminal_ and _stateful_.

**public inline fun <T, R : Any, C : MutableCollection<R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {**

    var sequence = doubleBufferedTransformingIndexedSequence镛(target: C)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element and the element itself
    * and returns the result of the transform applied to the element.
    * The operation is _terminal_ and _stateful_.

**public fun <T> Sequence<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {**

    var sequence = doubleBufferedTransformingIndexedSequence镛(target: C)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element and the element itself
    * and returns the result of the transform applied to the element.
    * The operation is _terminal_ and _stateful_.

**public fun <T, R : Any> Sequence<T>.mapNotNull(transform: (T) -> R?): Sequence<R> {**

    var sequence = doubleBufferedTransformingIndexedSequence镛(target: R)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element of the original sequence
    * and appends only the non-null results to the given destination.
    * The operation is _terminal_ and _stateful_.

**public inline fun <T, R : Any, C : MutableCollection<R>> Sequence<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {**

    var sequence = doubleBufferedTransformingIndexedSequence镛(target: C)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element of the original sequence
    * and appends only the non-null results to the given destination.
    * The operation is _terminal_ and _stateful_.

**public fun <T> Sequence<T>.withIndex(): Sequence<IndexedValue<T>> {**

    var sequence = doubleBufferedIndexedSequence镛(target: C)
    [:] sequence.
    filterNotNull().toListOrNull()
    * Applies the given [transform] function to each element of the original sequence
    * and appends only the non-null results to the given destination.
    * The operation is _terminal_ and _stateful_.

**public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {**

    var set = LinkedHashSet<T>()
    [:] sequence.
    filterNotNull().toListOrNull()
    * Returns a new [MutableSet] containing all distinct elements from the given sequence
    * having distinct keys returned by the given [selector] function.
    * The operation is _terminal_ and _stateful_.

**public inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {**

    var result = true
    [:] sequence.
    filterNotNull().toListOrNull()
    * Returns `true` if all elements match the given [predicate]
    * and returns `false` otherwise.
    * The operation is _terminal_ and _stateful_.

}
/**
 * Performs the given [action] on each element.
 *
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this)
        action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the
element.
 *
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 *
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(checkIndexOverflow(index++), item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.1")
public fun <T : Comparable<T>> Sequence<T>.max(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    if (!iterator.hasNext()) return maxElem
    var maxValue = selector(maxElem)
    do {
        val e = iterator.next()<@
            val v = selector(e)<@
            if (maxValue < v) {
                maxElem = e
                maxValue = v
            }
        } while (iterator.hasNext())
    return maxElem
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence.
 *
 * If any of values produced by [selector] function is `NaN`, the returned result
 * is `NaN`.
 *
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOf(selector: (T) -> Float): Float {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Sequence<T>.maxOf(selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.maxOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Sequence<T>.maxOfOrNull(selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R> Sequence<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R : Comparable<R>> Sequence<T>.maxOfWith(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R>
Sequence<T>.maxOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the sequence or `null` if there are no elements.
 * The operation is _terminal_.
 */

@SinceKotlin("1.4")
public fun Sequence<Double>.maxOrNull(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        max = maxOf(max, e)
    }
    return max
}

@SinceKotlin("1.4")
public fun Sequence<Float>.maxOrNull(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        max = maxOf(max, e)
    }
    return max
}

@SinceKotlin("1.4")
public fun <T : Comparable<T>> Sequence<T>.maxOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    if (!iterator.hasNext()) return maxElem
    var minValue = selector(maxElem)
    do {
        val e = iterator.next()
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    } while (iterator.hasNext())
    return minElem
}

@SinceKotlin("1.4")
public fun <T> Sequence<T>.maxWithOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(maxValue, e) < 0) {
            maxValue = e
        }
    }
    return maxValue
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * The operation is _terminal_.
 */

@SinceKotlin("1.4")
public fun <T, R : Comparable<R>> Sequence<T>.minByOrNull(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (minValue > e) {
            minElem = e
            minValue = e
        }
    }
    return minElem
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the sequence or `null` if any of values produced by [selector] function is `NaN`, the returned result
is `NaN`.

@throws NoSuchElementException if the sequence is empty.

The operation is _terminal_.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@Throws(NoSuchElementException::class)
public inline fun <T> Sequence<T>.minOf(selector: (T) -> Double): Double {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@Throws(NoSuchElementException::class)
public inline fun <T> Sequence<T>.minOf(selector: (T) -> Float): Float {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@Throws(NoSuchElementException::class)
public inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Double): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the sequence or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@Throws(NoSuchElementException::class)
public inline fun <T> Sequence<T>.minOfOrNull(selector: (T) -> Float): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```
/**
 * Returns the smallest value according to the provided [comparator]
 * among all values produced by [selector] function.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
@Throws(NoSuchElementException::class)
public inline fun <T, R : Comparable<R>> Sequence<T>.minOf(comparator: (T, R) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = comparator(iterator.next(), comparator(null, null))
    while (iterator.hasNext()) {
        val v = comparator(iterator.next(), comparator(null, null))
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R>
Sequence<T>.minOfWith(comparator: Comparator<in R>, selector: (T) -> R): R {
    val iterator = iterator()
    if (!iterator.hasNext()) throw NoSuchElementException()
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value according to the
 * provided [comparator] among all values produced by [selector] function applied to each element in the sequence.
 * 
 * The operation is _terminal_.
 * 
 * @return the smallest element or `null` if there are no elements.
 * @throws NoSuchElementException if the sequence is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@kotlin.internal.InlineOnly
public inline fun <T, R>
Sequence<T>.minOfWithOrNull(comparator: Comparator<in R>, selector: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minValue = selector(iterator.next())
    while (iterator.hasNext()) {
        val v = selector(iterator.next())
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest element or `null` if there are
 * no elements.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
public fun Sequence<Double>.minOrNull(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are
 * no elements.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
public fun Sequence<Float>.minOrNull(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        min = minOf(min, e)
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are
 * no elements.
 * 
 * The operation is _terminal_.
 */
@SinceKotlin("1.4")
public fun <T : Comparable<T>> Sequence<T>.minOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.",
ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun <T> Sequence<T>.minWith(comparator: Comparator<in T>): T? {
    return minWithOrNull(comparator)
}

/**
 * Returns the first element having the smallest value according to the
 * provided [comparator] or `null` if there are no elements.
 * 
 * The operation is _terminal_.
 * 
 * @return the first element having the smallest value or `null` if there are no elements.
 * @throws NoSuchElementException if the sequence is empty.
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.minWithOrNull(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the sequence has no elements.
 * 
 * The operation is _terminal_.
 * 
 * @return true if the sequence has no elements.
 */
public fun <T> Sequence<T>.none(): Boolean {
    return !iterator().hasNext()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * The operation is _terminal_.
 * 
 * @param predicate the predicate to test each element of the sequence
 * @return true if no elements match the given [predicate].
 */
public inline fun <T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.1")
public fun <T> Sequence<T>.onEach(action: (T) -> Unit): Sequence<T> {
    return map {
        action(it)
        it
    }
}

/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.onEachIndexed(action: (index: Int, T) -> Unit): Sequence<T> {
    return mapIndexed { index, element ->
        action(index, element)
        element
    }
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.reduceIndexed(operation: (accumulator: T, T) -> T): T {
    return reduceIndexed { index, element ->
        operation(accumulator, element)
    }
}

/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.reduceNull(operation: (T, T) -> T): T {
    return reduceNull { it
}

/**
 * Returns a sequence which performs the given [action] on each element of the original sequence as they pass through it and applies [operation] from left to right
 * to current accumulator value and each element.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@SinceKotlin("1.4")
public fun <T> Sequence<T>.reduceWithNull(operation: (T, T) -> T): T {
    return reduceWithNull { it
}
receiver is empty.

```kotlin
public inline fun <S, T : S> Sequence<T>.reduce(operation: (acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.

    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}

public inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.

    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}

@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.runningFold(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var accumulator = initial
        for (element in this@runningFold) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
}
```

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R> Sequence<T>.runningFoldNull(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var accumulator = initial
        for (element in this@runningFold) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
}
```

receiver is empty.

```kotlin
*param [operation] function that takes current accumulator value and an element,
and calculates the next accumulator value.
The operation is _terminal_.

*sample
```

```kotlin
samples.collections.Collections.Aggregates.reduce
```
(or should not be mutated)* as it may be passed to [operation] function later because of sequence’s lazy nature.

* @param [operation] function that takes the index of an element, current accumulator value\n and the element itself, and calculates the next accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.collections.Collections.Aggregates.runningFold\n

public fun <T, R> Sequence<T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    var accumulator = operation(checkIndexOverflow(index++), initial, element)\n    var index = 0\n    var accumulator = initial\n    return sequence {
        yield(accumulator)\n        accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n        yield(accumulator)\n        index = index + 1\n        accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n        yield(accumulator)\n    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.\n
* @param [operation] function that takes an index of an element, current accumulator value\n and the element itself, and calculates the next accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.collections.Collections.Aggregates.runningFold

 */
@SinceKotlin("1.4")
public fun <T, R> Sequence<T>.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)\n        var index = 0\n        var accumulator = initial\n        for (element in this@runningFoldIndexed) {
            accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n            yield(accumulator)\n            index = index + 1\n            accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n            yield(accumulator)\n        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this sequence.\n
* @param [operation] function that takes current accumulator value and the element, and calculates the next\n accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.Collections.Aggregates.runningFold

 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <S, T : S> Sequence<T>.runningReduce(operation: (acc: S, T) -> S): Sequence<S> {
    val iterator = iterator()\n    var accumulator: S = iterator.next()\n    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())\n        yield(accumulator)\n    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.\n
* @param [operation] function that takes the index of an element, current accumulator value\n and the element itself, and calculates the next accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.collections.Collections.Aggregates.runningReduce

 */
@SinceKotlin("1.4")
public fun <S, T : S> Sequence<T>.runningReduceIndexed(operation: (index: Int, acc: S, T) -> S): Sequence<S> {
    val iterator = iterator()\n    var accumulator: S = iterator.next()\n    var index = 1\n    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())\n        yield(accumulator)\n        index = index + 1\n    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this sequence.\n
* @param [operation] function that takes current accumulator value and an element, and calculates the next\n accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.Collections.Aggregates.scan

 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R> Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    var accumulator = operation(checkIndexOverflow(index++), initial, element)\n    return sequence {
        yield(accumulator)\n        accumulator = operation(checkIndexOverflow(index++), accumulator, element)\n        yield(accumulator)\n    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right\n * to each element and current accumulator value that starts with the first element of this sequence.\n
* @param [operation] function that takes the index of an element, current accumulator value\n and the element itself, and calculates the next accumulator value.\n
* The operation is _intermediate_ and _stateless_.\n
* @sample samples.Collections.Aggregates.scan

 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun <T, R>
Sequence<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return runningFoldIndexed(initial, operation)
}

Sequence<T>.scanReduce(operation: (acc: S, T) -> S): Sequence<S> {
    return runningReduce(operation)
}

Sequence<T>.scanIndexed(operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return runningFoldIndexed(operation)
}

Sequence<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): Sequence<S> {
    return runningReduceIndexed(operation)
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Double): Double {
    var sum: Double = 0.toDouble()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumOf(selector: (T) -> ULong): ULong {
    var sum: ULong = 0.toULong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T : Any> Sequence<T?>.requireNoNulls(): Sequence<T> {
    return map { it ?: throw IllegalArgumentException("null element found in $this.") }
}

/**
 * Splits this sequence into a sequence of lists each not exceeding the given [size].
 * The last list in the resulting sequence may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in
The operation is _intermediate_ and _stateful_.

```kotlin
@SinceKotlin("1.2")
public fun <T>
    Sequence<T>.chunked(size: Int): Sequence<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this sequence into several lists each not exceeding the given [size] and applies the given [transform] function to each.
 * @return sequence of results of the [transform] applied to each list.
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last list may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.
 * The operation is _intermediate_ and _stateful_.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            var removed = false
            return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements].
 * Note that the source sequence and the array being subtracted are iterated only when an ‘iterator’ is requested from the resulting sequence.
 * Changing any of them between successive calls to ‘iterator’ may affect the result.
 * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @return sequence of results of the [transform] applied to each list.
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it. The last list may have less elements than the given [size].
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this sequence.
 * The operation is _intermediate_ and _stateful_.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {
    if (elements.isEmpty()) return this
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] sequence.
 * The operation is _intermediate_ for this sequence and _terminal_ and _stateful_ for the [elements] sequence.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @return sequence of results of the [transform] applied to each list.
 * Note that the source sequence and the sequence being subtracted are iterated only when an ‘iterator’ is requested from the resulting sequence.
 * Changing any of them between successive calls to ‘iterator’ may affect the result.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] collection.
 * Note that the source sequence and the collection being subtracted are iterated only when an ‘iterator’ is requested from the resulting sequence.
 * Changing any of them between successive calls to ‘iterator’ may affect the result.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @return sequence of results of the [transform] applied to each list.
 * Note that the source sequence and the sequence being subtracted are iterated only when an ‘iterator’ is requested from the resulting sequence.
 * Changing any of them between successive calls to ‘iterator’ may affect the result.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.convertToSetForSetOperation()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] sequence.
 * The operation is _intermediate_ for this sequence and _terminal_ and _stateful_ for the [elements] sequence.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @return sequence of results of the [transform] applied to each list.
 * Note that the source sequence and the sequence being subtracted are iterated only when an ‘iterator’ is requested from the resulting sequence.
 * Changing any of them between successive calls to ‘iterator’ may affect the result.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of ‘hashCode()’ that doesn’t change between successive invocations.
 * The operation is _intermediate_ and _stateful_.
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements].
 * The operation is _intermediate_ and _stateful_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> {
    return minus(element)
}
```
elements for which [predicate] yielded `true`,\n  * while *second* list contains elements for which [predicate] yielded `false`.\n  *\n  * The operation is _terminal_.\n  *\n  * @sample\n
```kotlin
samples.collections.Sequences.Transformations.partition\n
/**\n * Returns a sequence containing all elements of the original sequence and then the given [element].\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic inline fun <T,\n
Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}

/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] array.\n * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(element: T): Sequence<T> {\n    return sequenceOf(this, sequenceOf(element)).flatten()\n}

/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] collection.\n * Note that the source sequence and the collection being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(elements: Array<out T>): Sequence<T> {\n    return this.plus(elements.asList())\n}

/**\n * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] sequence.\n * Note that the source sequence and the sequence being added are iterated only when an `iterator` is requested from\n * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic operator fun <T> Sequence<T>.plus(elements: Sequence<T>): Sequence<T> {\n    return sequenceOf(this, elements).flatten()\n}

/**\n * Returns a sequence containing all elements of original sequence and then the given [element].\n *\n * The operation is _intermediate_.\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {\n    return plus(element)\n}

/**\n * Returns a sequence of snapshots of the window of the given [size] sliding\n * along this sequence with the given [step], where each\n * snapshot is a list.\n *\n * Several last lists may have less elements than the given [size].\n * Both [size] and [step] must be positive and can be greater than the number of\n * elements in this sequence.\n *\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on each step, by default 1\n *\n * @sample samples.collections.Sequences.Transformations.takeWindows\n */\n@SinceKotlin("1.2")\npublic fun <T> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<List<T>> {\n    return windowedSequence(size, step, partialWindows, reuseBuffer = false)\n}

/**\n * Returns a sequence of results of applying the given [transform] function to\n * an each list representing a view over the window of the given [size] sliding\n * along this sequence with the given [step].\n *\n * @param size the number of elements to take in each window\n * @param step the number of elements to move the window forward by on each step, by default 1\n * @param partialWindows controls whether or not to keep partial windows in the end if any,\n * by default `false` which means partial windows won't be preserved\n *\n * @sample samples.collections.Sequences.Transformations.averageWindows\n */\n@SinceKotlin("1.2")\npublic fun <T, R> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R): Sequence<R> {\n    return windowedSequence(size, step, partialWindows, reuseBuffer = true).map(transform)\n}
The resulting sequence ends as soon as the shortest input sequence ends.

*The operation is intermediate and stateless.*

@sample

collections.Sequences.Transformations.zip

/**
 * Returns a sequence of values built from the elements of `this` sequence and the [other] sequence with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The resulting sequence ends as soon as the shortest input sequence ends.
 * The operation is intermediate and stateless.
 * @sample

```kotlin
public infix fun <T, R> Sequence<T>.zip(other: Sequence<R>): Sequence<Pair<T, R>> {
    return MergingSequence(this, other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a sequence of pairs of each two adjacent elements in this sequence if this sequence contains less than two elements.
 * The operation is intermediate and stateless.
 * @sample
collections.Sequences.Transformations.zipWithNext

```kotlin
/**
 * Returns a sequence containing the results of applying the given [transform] function
 * to an each pair of two adjacent elements in this sequence.
 * The returned sequence is empty if this sequence contains less than two elements.
 * The operation is intermediate and stateless.
 * @sample
collections.Sequences.Transformations.zipWithNextToFindDeltas

```kotlin
/**
 * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.
 * The operation is terminal.
 * @sample
collections.Sequences.joinTo

```kotlin
/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * The operation is terminal.
 * @sample
collections.Sequences.joinToString

```kotlin
/**
 * Returns this sequence as a [Sequence].
 * @kotlin.jvm:inline
 * @sample
collections.Sequences.asSequence

```kotlin
/**
 * Returns an average value of elements in the sequence.
 * The operation is terminal.
 * @kotlin.jvm:JvmName("averageOfByte")
```
Double = 0.0
var count: Int = 0
for (element in this) {
    sum += element
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Sequence<Double>.sum(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Sequence<Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Sequence<Double>.sum(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Sequence<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Sequence<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Sequence<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Sequence<Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Sequence<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Sequence<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Sequence<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Sequence<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Sequence<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
* Returns a set containing all elements of the original set except the elements contained in the given [elements] collection. The returned set preserves the element iteration order of the original set.

```kotlin
operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    val other = elements.convertToSetForSetOperationWith(this)
    if (other.isEmpty())
        return this.toSet()
    if (other is Set)
        return this.filterNotTo(LinkedHashSet<T>()) { it in other }
    val result = LinkedHashSet<T>(this)
    result.removeAll(other)
    return result
}
```

```kotlin
/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 * @sample samples.collections.Collections.Elements.minus
 */
public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(size * 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}
```

```kotlin
/**
 * Returns a set containing all elements of the original set and the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * @sample samples.collections.Collections.Elements.plus
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}
```

```kotlin
/**
 * Returns a set containing all elements of the original set and the given [elements] array,
 * which aren't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * @sample samples.collections.Collections.Elements.plus
 */
public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(size + elements.size))
    result.addAll(this)
    result.addAll(elements)
    return result
}
```

```kotlin
/**
 * Returns a set containing all elements of the original set and the given [elements] collection,
 * which aren't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * @sample samples.collections.Collections.Elements.plus
 */
public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(elements.collectionSizeOrNull()?.let { this.size + it } ?: this.size * 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}
```

```kotlin
/**
 * Returns a set containing all elements of the original set and the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * @sample samples.collections.Collections.Elements.plus
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}
```

```kotlin
/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 * @sample samples.collections.Collections.CharSequence.elementAt
 */
public expect fun CharSequence.elementAt(index: Int): Char
```

```kotlin
/**
 * Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.
 * @sample samples.collections.Collections.CharSequence.elementAtOrElse
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
```

```kotlin
/**
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * @sample samples.collections.Collections.CharSequence.elementAt
 */
public expect fun CharSequence.elementAt(index: Int): Char
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains
 * s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt file. See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

package kotlin.text

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt file. See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

// Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
samples.collections.Collections.Elements.elementAtOrNull
 */
@kotlin.internal.InlineOnly
public inline fun
CharSequence.elementAtOrNull(index: Int): Char? {
    return this.getOrNull(index)
}

/**
 * Returns the first character matching the given [predicate], or `null` if no such character was found.
 */
@kotlin.internal.InlineOnly
public inline fun
CharSequence.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
}

/**
 * Returns first character.
 * @throws [NoSuchElementException] if the char sequence is empty.
 */
public fun
CharSequence.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Char sequence is empty.")
    return this[0]
}

/**
 * Returns the first character matching the given [predicate].
 * @throws [NoSuchElementException] if no such character is found.
 */
public inline fun
CharSequence.firstOrNull(predicate: (Char) -> Boolean): Char? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Char sequence contains no character matching the predicate.")
}

/**
 * Returns a random character from this char sequence.
 * @throws NoSuchElementException if this char sequence is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun
CharSequence.random(): Char {
    return random(Random)
}

/**
 * Returns a random character from this char sequence using the specified source of randomness.
 */
@SinceKotlin("1.3")
public fun
CharSequence.random(random: Random): Char {
    if (isEmpty())
        throw NoSuchElementException("Char sequence is empty.")
    return get(random.nextInt(length))
}
Returns a random character from this char sequence, or `null` if this char sequence is empty.

```kotlin
*$/n/@SinceKotlin(\"1.4\")$/n/@WasExperimental(ExperimentalStdalibApi::class)/n/@WasExperimental(ExperimentalStdalibApi::class)/n/public inline fun CharSequence.randomOrNull(): Char? { /n  \n  return randomOrNull(Random) \n} /n/**
*\n*/
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
public inline fun CharSetquence.randomOrNull(): Char? {
    return randomOrNull(Random)
}

/**
 * Returns a random character from this char sequence using the specified source of randomness, or `null` if this char sequence is empty or has more than one character.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun CharSequence.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return get(random.nextInt(length))
}

/**
 * Returns the single character, or throws an exception if the char sequence is empty or has more than one character.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun CharSequence.single(): Char {
    return when (length)
    { 0 -> throw NoSuchElementException("Char sequence is empty.") 1 -> this[0] } else null
}

/**
 * Returns the single character matching the given [predicate], or throws exception if there is no or more than one matching character.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public fun CharSequence.singleOrNullable(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw
                IllegalArgumentException("Char sequence contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw
        NoSuchElementException("Char sequence contains no matching character.")
    @Suppress("UNCHECKED_CAST")
    return single as Char
}

/**
 * Returns the single character matching the given [predicate], or `null` if character was not found or more than one character was found.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun CharSequence.singleOrNull(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
}

/**
 * Returns a subsequence of this char sequence with the first [n] characters removed.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.text.Strings.drop
 */
public fun CharSequence.drop(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(n.coerceAtMost(length), length)
}

/**
 * Returns a string with the first [n] characters removed.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.text.Strings.drop
 */
public fun String.drop(n: Int): String {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return substring(n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing all characters except last characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return subSequence(0, index + 1)
    return ""
}

/**
 * Returns a string containing all characters except last characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return substring(0, index + 1)
    return ""
}
```
for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in this.indices) if (!predicate(this[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)

for (index in thisindices) if (!predicatethis[index])) return substring(index)
CharSequence.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
    keySelector, valueTransform)\n}

AssociateByToMap inherits from Map, which means that the return value of the function will be a Map<K, V>.

**CharSequence.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `transform`: A function that takes a Char and returns a Pair<K, V>.

This function will add all elements of this char sequence to the provided destination map, with keys and values transformed by the `transform` function.

**Example**

```kotlin
CharSequence.associateTo(destination: LinkedHashMap<Int, String>(), transform = { it -> Pair(it, it.toString()) })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `transform` function applied to each character of the given char sequence.

---

**CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V>**

- `valueSelector`: A function that takes a Char and returns a V.

This function will create a new Map with keys as Char and values as V, where each key-value pair is derived from the `valueSelector` function applied to each character of the given char sequence.

**Example**

```kotlin
CharSequence.associateWith(valueSelector = { it -> it.toString() })
```

This will create a new Map with keys as Char and values as String, where each key-value pair is derived from the `valueSelector` function applied to each character of the given char sequence.

---

**CharSequence.toCollection(destination: C): C**

- `destination`: The collection to which the elements of this char sequence will be added.

This function will add all elements of this char sequence to the provided collection.

**Example**

```kotlin
CharSequence.toCollection(ArrayList<String>())
```

This will create a new ArrayList with all elements of the given char sequence.

---

**CharSequence.toHashSet(): Set<Char>**

This function will create a new HashSet with all elements of this char sequence.

**Example**

```kotlin
CharSequence.toHashSet()
```

This will create a new HashSet with all elements of the given char sequence.

---

**CharSequence.toMutableList(): MutableList<Char>**

This function will create a new MutableList with all elements of this char sequence.

**Example**

```kotlin
CharSequence.toMutableList()
```

This will create a new MutableList with all elements of the given char sequence.

---

**CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R>**

- `transform`: A function that takes a Char and returns an Iterable<R>.

This function will create a new List with all elements of this char sequence, where each element is obtained by applying the `transform` function to each character of the given char sequence.

**Example**

```kotlin
CharSequence.flatMap { it -> List<String>(it.toString()) }
```

This will create a new List with all elements of the given char sequence, where each element is obtained by converting each character to a String.

---

**CharSequence.associateWithTo(destination: M, keySelector: (Char) -> K): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `keySelector`: A function that takes a Char and returns a K.

This function will add all elements of this char sequence to the provided destination map, with keys derived from the `keySelector` function.

**Example**

```kotlin
CharSequence.associateWithTo(destination: LinkedHashMap<Int, String>(), keySelector = { it -> it.toInt() })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `keySelector` function applied to each character of the given char sequence.

---

**CharSequence.associateByTo(destination: M, keySelector: (Char) -> K): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `keySelector`: A function that takes a Char and returns a K.

This function will add all elements of this char sequence to the provided destination map, with keys derived from the `keySelector` function.

**Example**

```kotlin
CharSequence.associateByTo(destination: LinkedHashMap<Int, String>(), keySelector = { it -> it.toInt() })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `keySelector` function applied to each character of the given char sequence.

---

**CharSequence.associateByToWithValueTransform(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `keySelector`: A function that takes a Char and returns a K.
- `valueTransform`: A function that takes a Char and returns a V.

This function will add all elements of this char sequence to the provided destination map, with keys derived from the `keySelector` function and values derived from the `valueTransform` function.

**Example**

```kotlin
CharSequence.associateByToWithValueTransform(destination: LinkedHashMap<Int, String>(), keySelector = { it -> it.toInt() }, valueTransform = { it -> it.toString().toUpperCase() })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `keySelector` function applied to each character of the given char sequence, and the values are derived from the `valueTransform` function applied to the same characters.

---

**CharSequence.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `transform`: A function that takes a Char and returns a Pair<K, V>.

This function will add all elements of this char sequence to the provided destination map, with keys and values derived from the `transform` function applied to each character of the given char sequence.

**Example**

```kotlin
CharSequence.associateTo(destination = LinkedHashMap<Int, String>(), transform = { it -> Pair(it.toInt(), it.toString().toUpperCase()) })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `transform` function applied to each character of the given char sequence.

---

**CharSequence.associateWithToWithValueTransform(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `keySelector`: A function that takes a Char and returns a K.
- `valueTransform`: A function that takes a Char and returns a V.

This function will add all elements of this char sequence to the provided destination map, with keys derived from the `keySelector` function and values derived from the `valueTransform` function.

**Example**

```kotlin
CharSequence.associateWithToWithValueTransform(destination = LinkedHashMap<Int, String>(), keySelector = { it -> it.toInt() }, valueTransform = { it -> it.toString().toUpperCase() })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `keySelector` function applied to each character of the given char sequence, and the values are derived from the `valueTransform` function applied to the same characters.

---

**CharSequence.associateWithTo(destination: M, transform: (Char) -> Pair<K, V>): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `transform`: A function that takes a Char and returns a Pair<K, V>.

This function will add all elements of this char sequence to the provided destination map, with keys and values derived from the `transform` function applied to each character of the given char sequence.

**Example**

```kotlin
CharSequence.associateWithTo(destination = LinkedHashMap<Int, String>(), transform = { it -> Pair(it.toInt(), it.toString().toUpperCase()) })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `transform` function applied to each character of the given char sequence.

---

**CharSequence.associateWithToWithValueTransform(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M**

- `destination`: The mutable Map to which the key-value pairs will be added.
- `keySelector`: A function that takes a Char and returns a K.
- `valueTransform`: A function that takes a Char and returns a V.

This function will add all elements of this char sequence to the provided destination map, with keys derived from the `keySelector` function and values derived from the `valueTransform` function.

**Example**

```kotlin
CharSequence.associateWithToWithValueTransform(destination = LinkedHashMap<Int, String>(), keySelector = { it -> it.toInt() }, valueTransform = { it -> it.toString().toUpperCase() })
```

This will create a new LinkedHashMap with keys as Int and values as String, where each key-value pair is derived from the `keySelector` function applied to each character of the given char sequence, and the values are derived from the `valueTransform` function applied to the same characters.

---

**CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V>**

- `valueSelector`: A function that takes a Char and returns a V.

This function will create a new Map with keys as Char and values as V, where each key-value pair is derived from the `valueSelector` function applied to each character of the given char sequence.

**Example**

```kotlin
CharSequence.associateWith(valueSelector = { it -> it.toString().toUpperCase() })
```

This will create a new Map with keys as Char and values as String, where each key-value pair is derived from the `valueSelector` function applied to each character of the given char sequence.

---

**CharSequence.toHashSet(): Set<Char>**

This function will create a new HashSet with all elements of this char sequence.

**Example**

```kotlin
CharSequence.toHashSet()
```

This will create a new HashSet with all elements of the given char sequence.

---

**CharSequence.toMutableList(): MutableList<Char>**

This function will create a new MutableList with all elements of this char sequence.

**Example**

```kotlin
CharSequence.toMutableList()
```

This will create a new MutableList with all elements of the given char sequence.

---

**CharSequence.toSet(): Set<Char>**

This function will create a new Set with all elements of this char sequence.

**Example**

```kotlin
CharSequence.toSet()
```

This will create a new Set with all elements of the given char sequence.
transform)\n\n/**
 * Returns a single list of all elements yielded from results of [transform] function being
 * invoked on each character in * and its index in the original char sequence.
 * @sample
 * samples.collections.Collections.Transformations.flatMapIndexed
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexed")
@kotlin.internal.InlineOnly
public fun <R> CharSequence.flatMapIndexed(transform: (Int, Char) -> Iterable<R>): List<R> {
    return flatMapIndexedTo(ArrayList<R>(), transform)
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each character
 * and its index in the original char sequence, to the given [destination].
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ByLambdaReturnType
@kotlin.jvm.JvmName("flatMapIndexedTo")
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> CharSequence.flatMapIndexedTo(destination: C, transform: (Int, Char) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups characters of the original char sequence by the key returned by the given [keySelector] function
 * and puts to the [destination] map each group key associated with a list of corresponding characters.
 * For the returned map preserves the entry iteration order of the keys produced from the original char sequence.
 */
public inline fun <K, M : MutableMap<in K, MutableList<Char>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Char>() }
        list.add(element)
    }
    return destination
}

/**
 * Groups characters of the original char sequence by the key returned by the given [keySelector] function
 * and puts to the [destination] map each group key associated with a list of corresponding characters.
 * For the returned map preserves the entry iteration order of the keys produced from the original char sequence.
 */
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a char sequence to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each character.
 */
@SinceKotlin("1.1")
public inline fun <K> CharSequence.groupingBy(crossinline keySelector: (Char) -> K): Grouping<Char, K> {
    return object :
Grouping<Char, K> { override fun sourceIterator(): Iterator<Char> = this@groupingBy.iterator() 
 override fun keyOf(element: Char): K = keySelector(element) 
 @sample samples.text.Strings.map

 /**
  * Returns a list containing the results of applying the given [transform] function to each character in the original char sequence.
  * 
  * @param [transform] function that takes the index of a character and the character itself in * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R> CharSequence.mapIndexed(transform: (index: Int, Char) -> R): List<R> {
      return mapIndexedTo(ArrayList<R>(length), transform)
  }

  /**
  * Returns a list containing only the non-null results of applying the given [transform] function to each character and its index in the original char sequence.
  * 
  * @param [transform] function that takes the index of a character and the character itself in * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R : Any> CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {
      return mapIndexedNotNullTo(ArrayList<R>(), transform)
  }

  /**
  * Applies the given [transform] function to each character and its index in the original char sequence
  * and appends the results to the given [destination].
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) -> R?): C {
      var index = 0
      for (item in this)
          destination.add(transform(index++, item))
      return destination
  }

  /**
  * Applies the given [transform] function to each character of the original char sequence
  * and appends the results to the given [destination].
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R, C :MutableCollection<in R>> CharSequence.mapTo(destination: C, transform: (Char) -> R): C {
      for (item in this)
          destination.add(transform(item))
      return destination
  }

  /**
  * Returns a lazy [Iterable] that wraps each character of the original char sequence into an
  * [IndexedValue] containing the index of that character and the character itself.
  * 
  */
  public fun CharSequence.withIndex(): Iterable<IndexedValue<Char>> {
      return IndexingIterable { iterator() }
  }

  /**
  * Returns `true` if all characters match the given [predicate].
  * 
  * @sample samples.collections.Collections.Aggregates.all
  */
  public inline fun CharSequence.all(predicate: (Char) -> Boolean): Boolean {
      for (element in this) if (!predicate(element)) return false
      return true
  }

  /**
  * Returns `true` if char sequence has at least one character.
  * 
  * @sample samples.collections.Collections.Aggregates.any
  */
  public fun CharSequence.any(): Boolean {
      return !isEmpty()
  }

  /**
  * Returns `true` if at least one character matches the given [predicate].
  * 
  * @sample samples.collections.Collections.Aggregates.anyWithPredicate
  */
  public inline fun CharSequence.any(predicate: (Char) -> Boolean): Boolean {
      for (element in this) if (predicate(element)) return true
      return false
  }

  /**
  * Returns the length of this char sequence.
  * 
  */
  public inline fun CharSequence.length(): Int {
      return length
  }

  /**
  * Returns the number of characters matching the given
  * 
  */
  public inline fun CharSequence.count(): Int {
      return length
  }

  /**
  * Returns a list containing the results of applying the given [transform] function to each character in the original char sequence.
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R> CharSequence.mapIndexed(transform: (index: Int, Char) -> R): List<R> {
      return mapIndexedTo(ArrayList<R>(length), transform)
  }

  /**
  * Returns a list containing only the non-null results of applying the given [transform] function to each character and its index in the original char sequence.
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R : Any> CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?): List<R> {
      return mapIndexedNotNullTo(ArrayList<R>(), transform)
  }

  /**
  * Applies the given [transform] function to each character and its index in the original char sequence
  * and appends the results to the given [destination].
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) -> R?): C {
      var index = 0
      for (item in this)
          destination.add(transform(index++, item))
      return destination
  }

  /**
  * Applies the given [transform] function to each character of the original char sequence
  * and appends the results to the given [destination].
  * 
  * @param [transform] function that takes the index of a character and the character itself in *
  * and returns the results of the transform applied to the character.
  * 
  */
  public inline fun <R, C :MutableCollection<in R>> CharSequence.mapTo(destination: C, transform: (Char) -> R): C {
      for (item in this)
          destination.add(transform(item))
      return destination
  }

  /**
  * Returns a lazy [Iterable] that wraps each character of the original char sequence into an
  * [IndexedValue] containing the index of that character and the character itself.
  * 
  */
  public fun CharSequence.withIndex(): Iterable<IndexedValue<Char>> {
      return IndexingIterable { iterator() }
  }

  /**
  * Returns `true` if all characters match the given [predicate].
  * 
  * @sample samples.collections.Collections.Aggregates.all
  */
  public inline fun CharSequence.all(predicate: (Char) -> Boolean): Boolean {
      for (element in this) if (!predicate(element)) return false
      return true
  }

  /**
  * Returns `true` if char sequence has at least one character.
  * 
  * @sample samples.collections.Collections.Aggregates.any
  */
  public fun(CharSequence.any): Boolean {
      return !isEmpty()
  }

  /**
  * Returns `true` if at least one character matches the given [predicate].
  * 
  * @sample samples.collections.Collections.Aggregates.anyWithPredicate
  */
  public inline fun CharSequence.any(predicate: (Char) -> Boolean): Boolean {
      for (element in this) if (predicate(element)) return true
      return false
  }

  /**
  * Returns the length of this char sequence.
  * 
  */
  public inline fun CharSequence.length(): Int {
      return length
  }

  /**
  * Returns the number of characters matching the given
  * 
  */
  public inline fun CharSequence.count(): Int {
      return length
  }
public inline fun CharSequence.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to
 * left to each character and current accumulator value.
 * The returned result is 'NaN' if any of values produced by [operation] function is 'NaN'.
 * If there are no characters, the returned result is 'NaN'.
 * @param [operation] function that takes a character and current accumulator value, and
 * calculates the next accumulator value.
 * @param [initial] the character itself, and calculates the next accumulator value.
 */
public inline fun <R : Comparable<R>> CharSequence.maxOf(selector: (Char) -> R): R {
    var maxElem = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxElem = maxOf(maxElem, v)
    }
    return maxElem
}

/**
 * Performs the given [action] on each character.
 */
public inline fun CharSequence.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each character, providing sequential index with the character.
 * @param [action] function that takes the index of a character and the character itself and
 * performs the action on the character.
 */
public inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

@Deprecated("Use maxOrNull instead.", ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public fun CharSequence.max(): Char? {
    return maxOrNull()
}

@Deprecated("Use maxByOrNull instead.", ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun <R : Comparable<R>> CharSequence.maxBy(selector: (Char) -> R): Char? {
    return maxByOrNull(selector)
}

/**
 * Returns the first character yielding the largest value of the given function or
 * 'null' if there are no characters.
 * @param [selector] function that returns the maximum value.
 * @param [maxValue] reference to the maximum value.
 * @param [i] index of the character.
 * @param [maxElem] reference to the maximum element.
 * @param [lastIndex] last index of the sequence.
 * @param [val] value of the character.
 * @param [v] reference to the next maximum value.
 * @param [e] reference to the next element.
 * @param [maxValue] reference to the maximum value.
 * @return the first character yielding the largest value of the given function.
 */
public inline fun <R : Comparable<R>> CharSequence.maxOrNull(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Performs the given [action] on each character in the char sequence.
 * @param [selector] function that takes a character and current accumulator value, and
 * calculates the next accumulator value.
 * The returned result is 'NaN' if any of values produced by [selector] function is 'NaN'.
 * If there are no characters, the returned result is 'NaN'.
 */
public inline fun <R : Comparable<R>> CharSequence.foldRightIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, accumulator, get(index--))
    }
    return accumulator
}

/**
 * Performs the given [action] on each character and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Performs the given [action] on each character in the original char sequence.
 */
public inline fun <R> CharSequence.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Performs the given [action] on each character.
 */
public inline fun CharSequence.fold(initial: R, operation: (R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Performs the given [action] on each character in the original char sequence.
 */
public inline fun <R> CharSequence.foldIndexed(initial: R, operation: (Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(element)
    return accumulator
}

/**
 * Performs the given [action] on each character.
 */
public inline fun CharSequence.foldIndexed(initial: R, operation: (index: Int, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, element)
    return accumulator
}
maxValue

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun CharSequence.maxOf(selector: (Char) -> Float): Float
{
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.maxOf(selector: (Char) -> R): R
{
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun CharSequence.maxOfOrNull(selector: (Char) -> Double): Double?
{
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> CharSequence.maxOfOrNull(selector: (Char) -> R): R?
{
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each character in the char sequence.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun CharSequence.maxOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R
{
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function applied to each character in the char sequence or `null` if there are no characters.
 * @throws NoSuchElementException if the char sequence is empty.
 */
@SinceKotlin("1.4")@OptIn(kotlin.experimental.ExperimentalTypeInference::class)}@OverloadResolution
ByLambdaReturnType@kotlin.internal.InlineOnly
public inline fun <R> CharSequence.maxOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R?
{
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
Returns the largest character or `null` if there are no characters.

```kotlin
@.SinceKotlin("1.4")
public fun CharSequence.maxOrNull(): Char? { // @Deprecated("Use maxOrNull instead.")
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

@Deprecated("Use maxWithOrNull
instead.", ReplaceWith("maxWithOrNull(comparator)"))
@SinceKotlin(warningSince = "1.4")
public fun CharSequence.maxWith(comparator: Comparator<in Char>): Char? {
    @DeprecatedSinceKotlin(warningSince = "1.4")
    return maxWithOrNull(comparator)
}

@Deprecated("Use maxWithOrNull instead.", ReplaceWith("maxWithOrNull(comparator)"))
@SinceKotlin(warningSince = "1.4")
public fun CharSequence.maxWithOrNull(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@Deprecated("Use minOrNull instead.", ReplaceWith("minOrNull")
@SinceKotlin("1.4")
public fun CharSequence.min(): Char? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("minByOrNull(selector)")
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharSequence.minBy(selector: (Char) -> R): Char? {
    return minByOrNull(selector)
}

@Deprecated("Use minByOrNull instead.", ReplaceWith("minByOrNull(selector)")
@SinceKotlin("1.4")
public inline fun <R : Comparable<R>> CharSequence.minByOrNull(selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

- The `maxOrNull()` function returns the largest character in the sequence or `null` if the sequence is empty.
- The `maxWithOrNull()` function allows for a custom comparator to be used for comparison.
- The `min()` function returns the smallest character in the sequence.
- The `minBy()` function allows for a custom function to be used for comparison.
- The `minByOrNull()` function returns the smallest character in the sequence or `null` if the sequence is empty.

These functions are designed to work with the `CharSequence` interface, which is part of the Kotlin standard library, providing flexibility and convenience for working with character sequences in Android development.
each character in the char sequence or `null` if there are no characters.

If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly//\public inline fun CharSequence.minOfOrNull(selector: (Char) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly//\public inline fun <R : Comparable<R>>
CharSequence.minOfOrNull(selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly//\public inline fun <R> CharSequence.minOfWith(comparator: Comparator<in R>, selector: (Char) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly//\public inline fun <R> CharSequence.minOfWithOrNull(comparator: Comparator<in R>, selector: (Char) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@kotlin.internal.InlineOnly//\public fun CharSequence.minOrNull(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

@Deprecated("Use minWithOrNull instead.", ReplaceWith("minWithOrNull(comparator)"))
@DeprecatedSinceKotlin(warningSince = "1.4")//\public fun CharSequence.minWith(comparator: Comparator<in Char>): Char? {
    return minWithOrNull(comparator)
}

@SinceKotlin("1.4")//\public inline fun <S : CharSequence> S.onEachIndexed(action: (index: Int, Char) -> Unit): S {
    return apply { forEachIndexed(action) }
}

@SinceKotlin("1.1")//\public inline fun <S : CharSequence> S.onEach(action: (Char) -> Unit): S {
    return apply { forEach(action) }
}

@SinceKotlin("1.4")//\public inline fun <S : CharSequence> S.onEachIndexed(action: (index: Int, Char) -> Unit): S {
    return apply { forEachIndexed(action) }
}
the first character and applying [operation] from left to right
* to current accumulator value and each character.
* Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,
* please use [reduceOrNull] instead. It returns `null` when its receiver is empty.
* @param [operation] function that takes current accumulator value and a character,
* and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty char sequence can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to
 * to current accumulator value and each character with its index in the original char sequence.
 * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,
 * please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of a character, current accumulator value and the character itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
CharSequence.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty char sequence can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first character and applying [operation] from left to
 * to current accumulator value and each character with its index in the original char sequence.
 * Returns `null` if the char sequence is empty.
 * @param [operation] function that takes the index of a character, current accumulator value and the character itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun
CharSequence.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the last character and applying [operation] from right to left
 * to each character and current accumulator value.
 * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes a character and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun
CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0)
        throw
UnsupportedOperationException("Empty char sequence can't be reduced.")
    var accumulator = get(index-)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}

/**
 * Accumulates value starting with the last character and applying [operation] from right to left
 * to each character with its index in the original char sequence and current accumulator value.
 * Throws an exception if this char sequence is empty. If the char sequence can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * @param [operation] function that takes the index of a character, the character itself and current accumulator value,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRightIndexed
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
public inline fun
CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
    var index
Open Source Used In DNAC Platform Arica 9679

```kotlin
public inline fun CharSequence.reduceRightIndexedOrNull(operation: (index: Int, Char, acc: Char) -> Char): Char?
{
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with the last character and applying [operation] from right to left to each character with its index in the original char sequence and current accumulator value.\n * \n * Returns `null` if the char sequence is empty.\n * \n * @param [operation] function that takes the index of a character, the character itself and current accumulator value,\n * \n * and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.4")
public inline fun CharSequence.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char?
{
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character and current accumulator value that starts with [initial] value.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningFold
 */
@SinceKotlin("1.4")
public inline fun <R> CharSequence.runningFold(initial: R, operation: (acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence.\n * \n * Note that `acc` value passed to [operation] function should not be mutated;\n * otherwise it would affect the previous value in resulting list.\n * \n * @param [operation] function that takes the index of a character, current accumulator value\n * \n * and the character itself, and calculates the next accumulator value.\n * \n * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
public inline fun CharSequence.runningReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply { add(accumulator) }
    for (index in 1 until length) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@sample

@sample samples.collections.Collections.Aggregates.runningReduce
**

@sample samples.collections.Collections.Aggregates.scan

@sample samples.collections.Collections.Aggregates.sumBy

@sample samples.collections.Collections.Aggregates.sumOf

@sample samples.collections.Collections.Aggregates.sumOfInt

@sample samples.collections.Collections.Aggregates.sumOfDouble

@sample samples.collections.Collections.Aggregates.sumOfInt

@sample samples.collections.Collections.Aggregates.sumOfDouble

@sample samples.collections.Collections.Aggregates.scan

@sample samples.collections.Collections.Aggregates.runningReduceIndexed
`@sinceKotlin("1.4")`@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution ByLambdaReturnType@kotlin.jvm.JvmName("sumOfLong")@kotlin.internal.InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> Long): Long {
    var sum = 0L
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

`@sinceKotlin("1.4")`@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution ByLambdaReturnType@kotlin.jvm.JvmName("sumOfUInt")@ExperimentalUnsignedTypes\n InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> UInt): UInt {
    var sum = 0U
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

`@sinceKotlin("1.4")`@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution ByLambdaReturnType@kotlin.jvm.JvmName("sumOfULong")@ExperimentalUnsignedTypes\n InlineOnly\n\npublic inline fun CharSequence.sumOf(selector: (Char) -> ULong): ULong {
    var sum = 0L
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

`@sinceKotlin("1.2")`\n\npublic fun CharSequence.chunked(size: Int): List<String> {
    return windowed(size, size, partialWindows = true)
}

`@sinceKotlin("1.2")`\n\npublic fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

`@sinceKotlin("1.2")`\n\npublic fun CharSequence.chunkedSequence(size: Int): Sequence<String> {
    return chunkedSequence(size) { it.toString() }
}

`@sinceKotlin("1.2")`\n\npublic fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}

`@sinceKotlin("1.2")`\n\npublic inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first, second)
}

`@sinceKotlin("1.2")`\n\npublic fun <R> CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first, second)
}
of strings, where *first* string contains characters for which [predicate] yielded 'true', while *second* string contains characters for which [predicate] yielded 'false'.

```kotlin
/*
 * Returns a list of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string. Several last strings may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any by default 'false' which means partial windows won't be preserved

@sample samples.collections.Sequences.Transformations.averageWindows
*/
@SinceKotlin("1.2")
public fun <R> CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {
    return windowedSequence(size, step, partialWindows) { it.toString() } // Returns a list of results of applying the given [transform] function to an each char sequence representing a view over the window of the given [size] sliding along this char sequence with the given [step]. Several last strings may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any by default 'false' which means partial windows won't be preserved

@sample samples.collections.Sequences.Transformations.averageWindows
*/
```

```kotlin
/*
 * Returns a list of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string. Several last strings may have less characters than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence. @param size the number of elements to take in each window @param step the number of elements to move the window forward by on each step, by default 1 @param partialWindows controls whether or not to keep partial windows in the end if any by default 'false' which means partial windows won't be preserved

@sample samples.collections.Sequences.Transformations.averageWindows
*/
```
@SinceKotlin("1.2")
public fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1,
    partialWindows: Boolean = false, transform: (CharSequence, Int, Int) -> R): Sequence<R> {
    checkWindowSizeStep(size, step)
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step step
        return windows.asSequence().map { index ->
            val start = index
            val end = index + size
            transform(this, start, end)
        }
}

/**
 * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index
 * using the provided `transform` function applied to each pair of characters.
 * The returned list has length of the shortest char sequence.
 * @sample samples.text.Strings.zipWithTransform
 */
public inline fun <V> CharSequence.zipWithTransform(other: CharSequence, transform: (CharSequence, V) -> V): List<Pair<Char, V>> {
    val length = minOf(this.length, other.length)
    val list = ArrayList<Pair<Char, V>>(length)
    for (i in 0 until length) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Creates an [Iterable] instance that wraps the original char sequence
 * returning its characters when being iterated.
 * @sample samples.collections.CollectionsITERABLE
 */
public fun CharSequence.asIterable(): Iterable<Char> {
    if (this is String && isEmpty()) return emptyList()
    return
}

/**
 * Creates a [Sequence] instance that wraps the original char sequence
 * returning its characters when being iterated.
 * @sample samples.collections.CollectionsSEQUENCE
 */
public fun CharSequence.asSequence(): Sequence<Char> {
    if (this is String && isEmpty()) return emptySequence()
    return
}

/**
 * Returns a sub sequence of this char sequence having
 * leading and trailing characters matching the [predicate] removed.
 */
public fun CharSequence.trim(predicate: (Char) -> Boolean): CharSequence {
    var startIndex = 0
    var endIndex = length
    var startFound = false

    while (startIndex <= endIndex) {
        val index = if (!startFound) startIndex
        else endIndex - 1
        val match = predicate(this[index])

        if (!startFound) {
            if (!match)
                startFound = true
            else
                startIndex += 1
        } else {
            if (!match)
                break
            else
                endIndex -= 1
        }
    }
    return subSequence(startIndex, endIndex)
}

/**
 * Returns a string having leading and trailing characters matching the [predicate] removed.
 */
public fun String.trim(predicate: (Char) -> Boolean): String {
    return (this as CharSequence).trim(predicate).toString()
}

/**
 * Returns a sub sequence of this char sequence having leading characters matching the [predicate] removed.
 */
public fun CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return"
}

/**
 * Returns a string having leading characters matching the [predicate] removed.
 */
public fun String.trimStart(predicate: (Char) -> Boolean): String {
    return (this as CharSequence).trimStart(predicate).toString()
}

/**
 * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.
 */
public fun CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return"
}

/**
 * Returns a string having trailing characters matching the [predicate] removed.
 */
public fun String.trimEnd(predicate: (Char) -> Boolean): String {
    return (this as CharSequence).trimEnd(predicate).toString()
}
Open Source Used In DNAC Platform Arica 9684

```kotlin
(Char) -> Boolean): CharSequence {\n    for (index in this.indices.reversed())\n        if (!predicate(this[index]))\n            return subSequence(0, index + 1)\n\n    return ""\n}

/**
 * Returns a string having trailing characters matching the [predicate] removed.\n */
public inline fun String.trim(predicate: (Char) -> Boolean): String =\n              trim { it in chars }\n
/**
 * Returns a string having leading and trailing characters from the [chars] array removed.\n */
public inline fun CharSequence.trimStart(vararg chars: Char): CharSequence =\n              trimStart { it in chars }\n
/**
 * Returns a string having leading characters from the [chars] array removed.\n */
public fun CharSequence.trimStart(vararg chars: Char):CharSequence =\n              trimStart { it in chars }\n
/**
 * Returns a string having trailing characters from the [chars] array removed.\n */
public fun CharSequence.trimEnd(vararg chars: Char):CharSequence =\n              trimEnd { it in chars }\n
/**
 * Returns a string having trailing whitespace removed.\n */
public fun CharSequence.trim(): CharSequence =\n              trim(Char::isWhitespace)\n
/**
 * Pads the string to the specified [length] with the specified character or space. Space is used by default.\n */
public fun String.padStart(length: Int, padChar: Char = ' '): String =\n              (this as CharSequence).padStart(length, padChar).toString()\n
/**
 * Pads the string to the specified [length] with the specified character or space.\n */
public fun CharSequence.padStart(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0)
        throw IllegalArgumentException("Desired length $length is less than zero.")\n    if (length <= this.length)
        return this.subSequence(0, this.length)\n
    val sb = StringBuilder(length)\n    for (i in 1..(length - this.length))
        sb.append(padChar)\n    sb.append(this)\n    return sb\n}
```

---

Open Source Used In DNAC Platform Arica 9684
@param length the desired string length.
* @param padChar the character to pad string with, if it has length less than the [length] specified. Space is used by default.
* @return Returns a string of length at least [length] consisting of `this` string appended with [padChar] as many times as are necessary to reach that length.
* @sample samples.text.Strings.padEnd

```kotlin
@public fun String.padEnd(length: Int, padChar: Char = ' '): String =
    (this as CharSequence).padEnd(length, padChar).toString()
```

```kotlin
/**
 * Returns `true` if this char sequence is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotEmpty(): Boolean = length > 0
```
[startIndex] and ending right before the [endIndex].

* @param startIndex the start index (inclusive).

* @param endIndex the end index (exclusive).

If not specified, the length of the char sequence is used.

@kotlin.internal.InlineOnly
public inline fun CharSequence.substring(startIndex: Int, endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()

* @kotlin.internal.InlineOnly
public inline fun CharSequence.substring(startIndex: Int, endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()

* @kotlin.internal.InlineOnly
public fun CharSequence.substring(range: IntRange): String = subSequence(range.start, range.endInclusive + 1).toString()

*public fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

*public fun String.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

*public fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

*public fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index, length)
}

*public fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    if (endIndex < startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb
}

*public fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String =
    (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()
given [range] in * is replaced with the [replacement] char sequence. \n * The end index of the [range] is included in the part to be replaced. \n * @param firstIndex the index of the first character to be removed. \n * @param endIndex the index of the first character after the removed part to keep in the string. \n * is not included in the removed part. \n */

```kotlin
@OptIn(ExperimentalStdlibApi::class)
public fun CharSequence.removeRange(startIndex: Int, endIndex: Int): CharSequence {
    if (endIndex < startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")

    if (endIndex == startIndex)
        return this.subSequence(0, length)

    val sb = StringBuilder(length - (endIndex - startIndex))
    sb.appendRange(this, 0, startIndex)
    sb.appendRange(this, endIndex, length)
    return sb.toString()
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun String.removeRange(startIndex: Int, endIndex: Int): String =
    (this as CharSequence).removeRange(startIndex, endIndex).toString()
```

```kotlin
/**
 * When this char sequence starts with the given [prefix] and ends with the given [suffix],
 * returns a new char sequence having both the given [prefix] and [suffix] removed.
 * Otherwise returns a new char sequence with the same characters. \n */
public fun CharSequence.removeSurrounding(prefix: CharSequence, suffix: CharSequence): CharSequence {
    if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {
        return subSequence(prefix.length, length - suffix.length)
    }
    return subSequence(0, length)
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun String.removeSurrounding(prefix: CharSequence, suffix: CharSequence): String {
    if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {
        return substring(prefix.length, length - suffix.length)
    }
    return this
}
```

```kotlin
@OptIn(ExperimentalStdlibApi::class)
public fun CharSequence.removeSurrounding(delimiter: CharSequence): CharSequence {
    if ((length >= delimiter.length + prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {
        return subSequence(delimiter.length, length - suffix.length - delimiter.length)
    }
    return subSequence(0, length)
}
```

```kotlin
@OptIn(ExperimentalStdlibApi::class)
public fun String.removeSurrounding(delimiter: CharSequence): String {
    if ((length >= delimiter.length + prefix.length + suffix.length) && startsWith(delimiter) && endsWith(delimiter)) {
        return this
    }
    return subSequence(0, length)
}
```
CharSequence = removeSurrounding(delimiter, delimiter) // Removes the given [delimiter] string from both the start and the end of this string
* If and only if it starts with and ends with the [delimiter].
* Otherwise returns this string unchanged.

//public fun String.removeSurrounding(delimiter: CharSequence): String
 = removeSurrounding(delimiter, delimiter) // Replace part of string before the first occurrence of given delimiter with the [replacement] string
*
* If the string does not contain the delimiter, returns
[missingDelimiterValue] which defaults to the original string.

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence): String =
    removeSurrounding(delimiter, delimiter)

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence): String =
    removeSurrounding(delimiter, delimiter) // Replace part of string before the first occurrence of given delimiter with the [replacement] string
*
* If the string does not contain the delimiter, returns
[missingDelimiterValue] which defaults to the original string.

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence): String =
    removeSurrounding(delimiter, delimiter) // Replace part of string before the first occurrence of given delimiter with the [replacement] string
*
* If the string does not contain the delimiter, returns
[missingDelimiterValue] which defaults to the original string.

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence): String =
    removeSurrounding(delimiter, delimiter) // Replace part of string before the first occurrence of given delimiter with the [replacement] string
*
* If the string does not contain the delimiter, returns
[missingDelimiterValue] which defaults to the original string.

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence): String =
    removeSurrounding(delimiter, delimiter) // Replace part of string before the first occurrence of given delimiter with the [replacement] string
*
@kotlin.internal.InlineOnly
public inline fun CharSequence.replace(regex: Regex, noinline transform: (MatchResult) -> CharSequence): String =
    regex.replace(this, transform)$n/*@param replacement A replacement expression that can include substitutions. See [Regex.replaceFirst] for
details.*/n/*@n*/@kotlin.internal.InlineOnly
public inline fun CharSequence.replaceFirst(regex: Regex, replacement: String): Boolean =
    regex.matches(this)$n/*Implementation of [regionMatches] for CharSequences.*/n/*Invoked when it's already known that arguments are not Strings, so that no additional type checks are performed.*/n/*Aninline fun CharSequence.regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean): Boolean =
    if (!ignoreCase && this is String && other is String)
        return this.regionMatches(other, 0, this.length - length, ignoreCase)
    else
        return regionMatchesImpl(0, other, 0, length, ignoreCase)$n/*public fun CharSequence.startsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this[0].equals(char, ignoreCase)$n/*Returns `true` if this char sequence starts with the specified character.*/n/*unpublic fun CharSequence.endsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this[this.length - 1].equals(char, ignoreCase)$n/*Returns `true` if this char sequence ends with the specified character.*/n/*public fun CharSequence.startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean =
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix, ignoreCase)$n/*else return*/n/*unpublic fun CharSequence.startsWith(prefix: CharSequence, startIndex: Int, ignoreCase: Boolean = false): Boolean =
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix, startIndex, ignoreCase)$n/*else return*/n/*public fun CharSequence.endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean =
    if (!ignoreCase && this is String && suffix is String)
        return this.endsWith(suffix, ignoreCase)$n/*else return*/n/*public fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val shortestLength = minOf(this.length, other.length)$n/*var i = 0 while (i < shortestLength &&
this[i].equals(other[i], ignoreCase = ignoreCase)) {*/n/*@param ignoreCase `true` to ignore character case when matching a character. By default `false`.*/n/*@sample samples.text.Strings.commonPrefixWith
*nunpublic fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val val shortestLength = minOf(this.length, other.length)$n/*var i = 0 while (i < shortestLength &&
this[i].equals(other[i], ignoreCase = ignoreCase)) {*/n/*@param ignoreCase `true` to ignore character case when matching a character. By default `false`.*/n/*@sample samples.text.Strings.commonPrefixWith
*nunpublic fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val val thisLength = this.length$nlval otherLength = other.length$nlval shortestLength = minOf(thisLength, otherLength)$n/*var i = 0 while (i < shortestLength &&
this[i].equals(other[i], ignoreCase = ignoreCase)) {*/n/*@param ignoreCase `true` to ignore character case when matching a character. By default `false`.*/n/*@sample samples.text.Strings.commonPrefixWith
*
string from [strings] or -1 if none of [strings] are found.

* To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from last to the beginning of this string, and finds at each position the first element in [strings] that matches this string at that position.

public fun CharSequence.indexOfAny(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int =
    findAnyOf(strings, startIndex, ignoreCase, last = false)?: -1

* @param ignoreCase `true` to ignore character case when matching a string. By default `false`.

* @return An index of the last occurrence of matched string from [strings] or -1 if none of [strings] are found.

* @param ignoreCase `true` to avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from the end toward the beginning of this string, and finds at each position the first element in [strings] that matches this string at that position.

public fun CharSequence.lastIndexOfAny(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int =
    findAnyOf(strings, startIndex, ignoreCase, last = true)?: -1

* @param ignoreCase `true` to ignore character case when matching a string. By default `false`.

* @return An index of the first occurrence of [char] or -1 if none is found.

* Anpublic fun CharSequence.indexOf(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this is String)
        indexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeIndexOf(char, startIndex)
}

* Anpublic fun CharSequence.lastIndexOf(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this is String)
        lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeLastIndexOf(char, startIndex)
}

* Anpublic fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =
    if (other is String)
        indexOf(other, ignoreCase) >= 0
    else
        indexOf(other, 0, length, ignoreCase) >= 0

* An@Suppress("INAPPLICABLE_OPERATOR_MODIFIER") public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =
    indexOf(other, ignoreCase) >= 0

* An@Suppress("INAPPLICABLE_OPERATOR_MODIFIER") public operator fun CharSequence.contains(char: Char, ignoreCase: Boolean = false): Boolean =
    indexOf(char, ignoreCase) >= 0
Open Source Used In DNAC Platform Arica 9692
rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more strings to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param * A to avoid ambiguous results when strings in [delimiters] have characters in common, this
 * method proceeds from the beginning to the end of this string, and matches at each position the first element in
 * [delimiters] that is equal to a delimiter in this instance at that position.
 * @return
 * @see
 */
public fun CharSequence.split(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): List<String> {
    if (delimiters.size == 1) {
        val delimiter = delimiters[0]
        return split(delimiter, ignoreCase, limit)
    }

    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }
}

/**
 * Splits this char sequence to a sequence of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more characters to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.
 * @param limit The maximum number of substrings to return.
 */
public fun CharSequence.splitToSequence(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0):
    Sequence<String> =
    rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more characters to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.
 * @param limit The maximum number of substrings to return.
 */
public fun CharSequence.split(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0):
    if (delimiters.size == 1) {
        return split(delimiters[0].toString(), ignoreCase, limit)
    }

    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiter].
 * This is specialized version of split which receives single non-empty delimiter and offers better performance.
 * @param delimiter String used as delimiter
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`.
 * @param limit The maximum number of substrings to return.
 */
private fun CharSequence.split(delimiter: String, ignoreCase: Boolean, limit: Int): List<String> {
    require(limit >= 0, "Limit must be non-negative, but was $limit." )

    var currentOffset = 0
    var nextIndex = indexOf(delimiter, currentOffset, ignoreCase)
    if (nextIndex == -1 || limit == 1) {
        return listOf(this.toString())
    }

    val isLimited = limit > 0
    val result = ArrayList<String>(if (isLimited) limit.coerceAtMost(10) else 10)
    do {
        result.add(substring(currentOffset, nextIndex))
        currentOffset = nextIndex + delimiter.length
        // Do not search for next occurrence if we're reaching limit.
        if (isLimited && result.size == limit - 1) break
        nextIndex = indexOf(delimiter, currentOffset, ignoreCase)
    } while (nextIndex != -1)

    result.add(substring(currentOffset, length))
    return result
}

/**
 * Splits this char sequence around matches of the given regular expression.
 * @param limit Non-negative value specifying the maximum number of substrings to return.
 * @return Zero by default means no limit is set.
 */
* An@kotlin.internal.InlineOnly
* public inline fun CharSequence.split(regex: RegExp, limit: Int = 0): List<String> =
* regex.split(this, limit).map { $1 }

* An@package kotlin.text
* public fun String.splitToSequence(regex: RegExp, limit: Int = 0): List<String> =
* regex.splitToSequence(this, limit)

*/

package kotlin.text

import kotlin.js.RegExp

/**
 * Converts the characters in the specified array to a string.
 * @param * A to avoid ambiguous results when strings in [delimiters] have characters in common, this
 * method proceeds from the beginning to the end of this string, and matches at each position the first element in
 * [delimiters] that is equal to a delimiter in this instance at that position.
 * @return
 */
@SinceKotlin("1.2")
@Deprecated("Use String.concatToString() instead", ReplaceWith("chars.concatToString()"))
@kotlin.internal.InlineOnly
public actual fun String(chars: CharArray): String {
    var result = ""
    for (char in chars) {
        result += char
    }
    return result
}
result

```kotlin
/**
 * Converts the characters from a portion of the specified array to a string.
 * @throws IndexOutOfBoundsException if either [offset] or [length] are less than zero
 * or `offset + length` is out of [chars] array bounds.
 * @throws IllegalArgumentException if [chars] is null.
 * @throws NullPointerException if [chars] is null.
 */
public actual fun String(chars: CharArray, offset: Int, length: Int): String {
    if (offset < 0 || length < 0 || offset + length > chars.size)
        throw IndexOutOfBoundsException("size: $chars.size; offset: $offset; length: $length")
    var result = ""
    for (index in offset until offset + length) {
        result += chars[index]
    }
    return result
}

/**
 * Concatenates characters in this [CharArray] into a String.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 */
public actual fun CharArray.concatToString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}

/**
 * Concatenates characters in this [CharArray] or its subrange into a String.
 * @param startIndex the beginning (inclusive) of the subrange of characters, 0 by default.
 * @param endIndex the end (exclusive) of the subrange of characters, size of this array by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 */
public actual fun CharArray.concatToString(startIndex: Int = 0, endIndex: Int = size): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, size)
    return CharArray(endIndex - startIndex) { get(startIndex + it) }
}

/**
 * Returns a [CharArray] containing characters of this string.
 */
public actual fun String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}

/**
 * Returns a [CharArray] containing characters of this string or its substring.
 * @param startIndex the beginning (inclusive) of the substring, 0 by default.
 * @param endIndex the end (exclusive) of the substring, length of this string by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
 */
public actual fun String.toCharArray(startIndex: Int = 0, endIndex: Int = this.length): CharArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return CharArray(endIndex - startIndex) { get(startIndex + it) }
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array.
 * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte sequence.
 */
public actual fun ByteArray.decodeToString(): String {
    return decodeUtf8(this, 0, size)
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array or its subrange.
 * @param startIndex the beginning (inclusive) of the subrange, 0 by default.
 * @param endIndex the end (exclusive) of the subrange, size of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement character \uFFFD.
 */
public actual fun ByteArray.decodeToString(startIndex: Int = 0, endIndex: Int = this.size, throwOnInvalidSequence: Boolean = false): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return decodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding.
 * @throws IllegalFormatException if the string contains malformed characters.
 */
public actual fun String.encodeToByteArray(): ByteArray {
    return encodeUtf8(this, 0, length)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding in this array or its subrange.
 * @param startIndex the beginning (inclusive) of the subrange, 0 by default.
 * @param endIndex the end (exclusive) of the subrange, size of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement character \uFFFD.
 */
public actual fun String.encodeToByteArray(startIndex: Int = 0, endIndex: Int = this.length, throwOnInvalidSequence: Boolean = false): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}
```

Open Source Used In DNAC Platform Arica 9694
open source used in DNAC platform Arica 9695

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
//auto-generated file. do not edit.

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UByteArray
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@PublishedApi
internal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> {
    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) {
    }
}

string or its substring to an array of bytes in UTF-8 encoding.

/* * @param startIndex the beginning (inclusive) of the substring to encode, 0 by default. * @param endIndex the end (exclusive) of the substring to encode, length of this string by default. * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char sequence or replace. */

* @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
* @throws IllegalArgumentEx ception if [startIndex] is greater than [endIndex].
* @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.
*/

@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.encodeToByteArray(
    startIndex: Int = 0,
    endIndex: Int = this.length,
    throwOnInvalidSequence: Boolean = false
): ByteArray {
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Returns a copy of this string converted to upper case using the rules of the default locale.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toUpperCase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to lower case using the rules of the default locale.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toLowerCase(): String = asDynamic().toLowerCase()

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeIndexOf(str: String, fromIndex: Int): Int = asDynamic().indexOf(str, fromIndex)

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeLastIndexOf(str: String, fromIndex: Int): Int = asDynamic().lastIndexOf(str, fromIndex)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int): String = asDynamic().substring(startIndex)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int, endIndex: Int): String = asDynamic().substring(startIndex, endIndex)

@kotlin.internal.InlineOnly
public inline fun String.concat(str: String): String = asDynamic().concat(str)

@kotlin.internal.InlineOnly
public inline fun String.match(regex: String): Array<String>? = asDynamic().match(regex)

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
public actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {
    if (ignoreCase) {
        val n1 = this.length
        val n2 = other.length
        val min = minOf(n1, n2)
        var start = 0
        while (true) {
            var end = minOf(start + 16, min)
            var s1 = this.substring(start, end)
            var s2 = other.substring(start, end)
            if (s1 != s2) {
                s1 = s1.toUpperCase()
                s2 = s2.toUpperCase()
                if (s1 != s2) {
                    s1 = s1.toLowerCase()
                    s2 = s2.toLowerCase()
                    return s1.compareTo(s2)
                }
            }
            if (end == min) break
            var s1 = s1.toUpperC ase()
            var s2 = s2.toUpperC ase()
            if (s1 != s2) {
                s1 = s1.toLowerCase()
                s2 = s2.toLowerCase()
                return s1.compareTo(s2)
            }
        }
    } else {
        return compareTo(other)
    }
}

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String> = STRING_CASE_INSENSITIVE_ORDER

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@kotlin.internal.InlineOnly
internal inline func
Open Source Used In DNAC Platform Arica  9696

this(ByteArray(size))

/**
 * Returns the array element at the given [index]. This method can be called using the index operator.
 * * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public operator fun get(index: Int): UByte = storage[index].toUByte()

/**
 * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
 * * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public operator fun set(index: Int, value: UByte) {
    storage[index] = value.toByte()
}

/** Returns the number of elements in the array. */
public override val size: Int get() = storage.size

/** Creates an iterator over the elements of the array. */
public override operator fun iterator(): UByteIterator =

private class Iterator(private val array: ByteArray) : UByteIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextUByte() = if (index < array.size)
        array[index++].toUByte() else throw NoSuchElementException(index.toString())
}

override fun contains(element: UByte): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    if ((element as Any?) !is UByte)
        return false
    return storage.contains(element.toByte())
}

override fun containsAll(elements: Collection<UByte>): Boolean {
    return (elements as Collection<*>).all { it is UByte &&
        storage.contains(it.toByte()) }
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray(size: Int, init: (Int) -> UByte): UByteArray {
    return UByteArray(ByteArray(size) { index ->
        init(index).toByte()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ubyteArrayOf(vararg elements: UByte): UByteArray = elements

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UIntArray

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@PublishedApi internal constructor(@PublishedApi internal val storage: IntArray) : Collection<UInt> {

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) :
        this(IntArray(size))

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     * * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
     * where the behavior is unspecified.
     */
    public operator fun get(index: Int): UInt = storage[index].toUInt()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
     * * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
     * where the behavior is unspecified.
     */
    public operator fun set(index: Int, value: UInt) {
        storage[index] = value.toInt()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): UIntIterator =

    private class Iterator(private val array: IntArray) : UIntIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUInt() = if (index < array.size)
            array[index++].toUInt() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UInt): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is UInt)
            return false
        return storage.contains(element.toInt())
    }

    override fun containsAll(elements: Collection<UInt>): Boolean {
        return (elements as Collection<*>).all { it is UInt &&
            storage.contains(it.toInt()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}
Creates a new array of the specified [size], where each element is calculated by calling the specified
function.

The function [init] is called for each array element sequentially starting from the first one. It should return the value for an array element given its index.

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
    return UIntArray(IntArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun uintArrayOf(vararg elements: UInt): UIntArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index -> init(index).toLong() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index -> init(index).toLong() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toInt() })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements
```
operator fun get(index: Int): UShort = storage[index].toUShort() /* Sets the element at the given [index] to the given [value]. This method can be called using the index operator. */

    */

    /**
     * Sets the element at the given index to the given value. This method can be called using the index operator.
     *
     * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
     * where the behavior is unspecified. * 
     */
    public operator fun set(index: Int, value: UShort) {
        storage[index] = value.toShort()
    }

    /**
     * Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): UShortIterator = Iterator(storage)

    private class Iterator(private val array: ShortArray) : UShortIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUShort() = if (index < array.size) array[index++].toUShort() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UShort): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed. Currently JS BE does not generate special bridge method for this method.
        // if ((element as Any?) != UShort)
        return false
    }

    override fun containsAll(elements: Collection<UShort>): Boolean {
        return (elements as Collection<*>).all { it is UShort && 
            storage.contains(it.toUShort()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function.
 * The function [init] is called for each array element sequentially starting from the first one. * 
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index ->
        init(index).toShort()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ushortArrayOf(vararg elements: UShort): UShortArray = elements
/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component1(): UInt {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component1(): ULong {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component1(): UByte {
    return get(0)
}

/**
 * Returns 1st *element* from the array.
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component1(): UShort {
    return get(0)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component2(): UInt {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component2(): ULong {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component2(): UByte {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component2(): UShort {
    return get(1)
}
the size of this array is less than 2, throws an [IllegalArgumentException] except in Kotlin/JS

/*
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component2(): ULong {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component2(): UByte {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component2(): UShort {
    return get(1)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component3(): UInt {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component3(): ULong {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component3(): UByte {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component3(): UShort {
    return get(2)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component4(): UInt {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component4(): ULong {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component4(): UByte {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component4(): UShort {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component5(): UInt {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component5(): ULong {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component5(): UByte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IllegalArgumentException] except in Kotlin/JS
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component5(): UShort {
    return get(4)
}
is unspecified.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component5(): ULong { return get(4) }\n```
Open Source Used In DNAC Platform Arica  9701

samples.collections.Collections.Elements.elementAtOrNull

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.elementAtOrNull(index: Int): ULong? { return this.getOrNull(index) } /* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrNull

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.elementAtOrNull(index: Int): UByte? { return this.getOrNull(index) } /* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrNull

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.elementAtOrNull(index: Int): UShort? { return this.getOrNull(index) } /* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrNull

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.find(predicate: (UInt) -> Boolean): UInt? { return firstOrNull(predicate) } /* Returns the first element matching the given [predicate], or `null` if no such element was found.

@sample samples.collections.Collections.Elements.find

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.find(predicate: (ULong) -> Boolean): ULong? { return firstOrNull(predicate) } /* Returns the first element matching the given [predicate], or `null` if no such element was found.

@sample samples.collections.Collections.Elements.find

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.find(predicate: (UByte) -> Boolean): UByte? { return firstOrNull(predicate) } /* Returns the first element matching the given [predicate], or `null` if no such element was found.

@sample samples.collections.Collections.Elements.find

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.find(predicate: (UShort) -> Boolean): UShort? { return firstOrNull(predicate) } /* Returns the first element matching the given [predicate], or `null` if no such element was found.

@sample samples.collections.Collections.Elements.find

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.first(): UInt { return storage.first().toUInt() } /* Returns first element.

@throws [NoSuchElementException] if the array is empty.

@sample samples.collections.Collections.Elements.first

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.first(): ULong { return storage.first().toUInt() } /* Returns first element.

@throws [NoSuchElementException] if the array is empty.

@sample samples.collections.Collections.Elements.first

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.first(): UByte { return storage.first().toUByte() } /* Returns first element.

@throws [NoSuchElementException] if the array is empty.

@sample samples.collections.Collections.Elements.first

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.first(): UShort { return storage.first().toUShort() } /* Returns first element.

@throws [NoSuchElementException] if the array is empty.

@sample samples.collections.Collections.Elements.first

*/
**Open Source Used In DNAC Platform Arica**

ULongArray.first(): ULong {n
    return storage.first().toULong();n
}n/*
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UByteArray.first(): UByte {n
    return storage.first().toUByte();n
}n/*
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UShortArray.first(): UShort {n
    return storage.first().toUShort();n
}n/*
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UIntArray.first(predicate: (UInt) -> Boolean): UInt {n
    for (element in this) if (predicate(element)) return
    throw NoSuchElementException("Array contains no element matching the predicate.");n
}n/*
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun ULongArray.first(predicate: (ULong) -> Boolean): ULong {n
    for (element in this) if (predicate(element)) return
    throw NoSuchElementException("Array contains no element matching the predicate.");n
}n/*
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UByteArray.first(predicate: (UByte) -> Boolean): UByte {n
    for (element in this) if (predicate(element)) return
    throw NoSuchElementException("Array contains no element matching the predicate.");n
}n/*
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UShortArray.firstOrNull(): Nullable<UInt> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element, or `null` if the array is empty.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun ULongArray.firstOrNull(): Nullable<ULong> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element, or `null` if the array is empty.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UByteArray.firstOrNull(): Nullable<UByte> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element, or `null` if the array is empty.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public inline fun UShortArray.firstOrNull(): Nullable<UShort> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element, or `null` if the array is not found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public Nullable<Nullable<UInt>> firstOrNull(predicate: (UInt) -> Boolean): Nullable<UInt> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public Nullable<Nullable<ULong>> firstOrNull(predicate: (ULong) -> Boolean): Nullable<ULong> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public Nullable<Nullable<UByte>> firstOrNull(predicate: (UByte) -> Boolean): Nullable<UByte> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */

*An@SinceKotlin("1.3")n@ExperimentalUnsignedTypes
public Nullable<Nullable<UShort>> firstOrNull(predicate: (UShort) -> Boolean): Nullable<UShort> {n
    return if (isEmpty()) null else this[0];n
}n/*
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? { 
    for (element in this) if (predicate(element)) return element
    return null
}  

/**
 * Returns an element at the given [index] or the result of calling the 
 * [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the 
 * [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the 
 * [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the 
 * [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {
    for (index in 0 until size) if (predicate(get(index))) return index
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {
    for (index in 0 until size) if (predicate(get(index))) return index
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {
    for (index in 0 until size) if (predicate(get(index))) return index
    return -1
}

/**
 * Returns first index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {
    for (index in 0 until size) if (predicate(get(index))) return index
    return -1
}
Returns index of the first element matching the given [predicate], or -1 if the array does not contain such element.

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int |
\nreturn storage.indexOfFirst {
\npredicate(it.toULong()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int |
\nreturn storage.indexOfFirst {
\npredicate(it.toUByte()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int |
\nreturn storage.indexOfFirst {
\npredicate(it.toUShort()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int |
\nreturn storage.indexOfLast {
\npredicate(it.toUInt()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int |
\nreturn storage.indexOfLast {
\npredicate(it.toULong()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int |
\nreturn storage.indexOfLast {
\npredicate(it.toUByte()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int |
\nreturn storage.indexOfLast {
\npredicate(it.toUShort()) }\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(): UInt |
\nreturn storage.last().toUInt()\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.last(): ULong |
\nreturn storage.last().toULong()\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.last(): UByte |
\nreturn storage.last().toUByte()\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.last(): UShort |
\nreturn storage.last().toUShort()\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.last(predicate: (UInt) -> Boolean): UInt |
\nfor (index in this.indices.reversed()) {
\nval element = this[index]
\nif (predicate(element)) return element
\n\nthrow NoSuchElementException("Array contains no element matching the predicate.")\n\n*/

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.last(predicate: (ULong) -> Boolean): ULong |
\nfor (index in this.indices.reversed()) {
\nval element = this[index]
\nif (predicate(element)) return element
\n\nthrow NoSuchElementException("Array contains no element matching the predicate.")\n\n*/
element matching the given \[predicate\].

* @throws [NoSuchElementException] if no such element is found.

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.last(predicate: (UByte) -> Boolean): UByte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.last(predicate: (USHort) -> Boolean): UShort {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf(element.toInt())
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.lastIndexOf(element: ULong): Int {
    return storage.lastIndexOf(element.toLong())
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf(element.toByte())
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.lastIndexOf(element: USHort): Int {
    return storage.lastIndexOf(element.toShort())
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.lastOrNull(): UInt? {
    return if (isEmpty()) null else this[size - 1]
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongArray.lastOrNull(): ULong? {
    return if (isEmpty()) null else this[size - 1]
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

* @SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.lastOrNull(): USHort? {
    return if (isEmpty()) null else this[size - 1]
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.random(): ULong {
    return random(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(): UByte? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array, or `null` if this array is empty.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(): UShort? {
    return randomOrNull(Random)
}
fun UByteArray.randomOrNull(random: Random): UByte? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness, or 'null' if this array is empty.
 */
fun UShortArray.randomOrNull(random: Random): UShort? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
fun UIntArray.single(): UInt {
    return storage.single().toUInt()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
fun ULongArray.single(): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as ULong
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UByte
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UShort
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.singleOrNull(): UInt? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.singleOrNull(): ULong? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.singleOrNull(predicate: (UByte) -> Boolean): UByte? {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.drop(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.drop(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.drop(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.drop(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}
n).coerceAtLeast(0))\n\n\n* Returns a list containing all elements except last [n] elements.

samples.Collections.Transformation.dropLast(n: Int): List<UInt>
{
    require(n > 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n\n* Returns a list containing all elements except last [n] elements.

samples.Collections.Transformation.dropLast(n: Int): List<ULong>
{
    require(n > 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n\n* Returns a list containing all elements except last [n] elements.

samples.Collections.Transformation.dropLast(n: Int): List<UByteArray>
{
    require(n > 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n\n* Returns a list containing all elements except last [n] elements.

{
    require(n > 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n\n* Returns a list containing all elements except last [n] elements.

samples.Collections.Transformation.dropWhile(predicate: (UInt) -> Boolean): List<UInt>
{
    var yielding = false
    val list = ArrayList<UInt>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

samples.Collections.Transformation.dropWhile(predicate: (UInt) -> Boolean): List<ULongArray>
{
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    return emptyList()
}

samples.Collections.Transformation.dropWhile(predicate: (ULong) -> Boolean): List<UByteArray>
{
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    return emptyList()
}

samples.Collections.Transformation.dropWhile(predicate: (UByteArray) -> Boolean): List<UByteArray>
{
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    return emptyList()
}

samples.Collections.Transformation.dropWhile(predicate: (UShortArray) -> Boolean): List<UShortArray>
{
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    return emptyList()
}

samples.Collections.Transformation.dropWhile(predicate: (List<UInt>) -> Boolean): List<UInt>
{
    var yielding = false
    val list = ArrayList<UInt>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

samples.Collections.Transformation.dropWhile(predicate: (List<ULong>) -> Boolean): List<ULong>
{
    var yielding = false
    val list = ArrayList<ULong>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

samples.Collections.Transformation.dropWhile(predicate: (List<UByteArray>) -> Boolean): List<UByteArray>
{
    var yielding = false
    val list = ArrayList<UByteArray>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

samples.Collections.Transformation.dropWhile(predicate: (List<UByteArray>) -> Boolean): List<UByteArray>
{
    var yielding = false
    val list = ArrayList<UByteArray>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

samples.Collections.Transformation.dropWhile(predicate: (List<UShortArray>) -> Boolean): List<UShortArray>
{
    var yielding = false
    val list = ArrayList<UShortArray>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
public inline fun ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {
    var yielding = false
    val list = ArrayList<ULong>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)){
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
public inline fun UByteArray.dropWhile(predicate: (UByte) -> Boolean): List<UByte> {
    var yielding = false
    val list = ArrayList<UByte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)){
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
public inline fun UShortArray.dropWhile(predicate: (UShort) -> Boolean): List<UShort> {
    var yielding = false
    val list = ArrayList<UShort>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)){
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.Collections.Filtering.filter
 */
public inline fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.Collections.Filtering.filter
 */
public inline fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.Collections.Filtering.filter
 */
public inline fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.Collections.Filtering.filter
 */
public inline fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * 
 * @sample samples.Collections.Filtering.filterIndexed
 */
public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {
    return filterIndexedTo(ArrayList<UInt>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * 
 * @sample samples.Collections.Filtering.filterIndexed
 */
public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(),
    predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 * 
 * @sample samples.Collections.Filtering.filterIndexed
 */
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(),
    predicate)
}
filterIndexedTo(ArrayList<UByte>(), predicate)\n\n\nReturns a list containing only elements matching the given [predicate].\n\n\n@param [predicate] function that takes the index of an element and the element itself\n\n\nand returns the result of predicate evaluation on the element.\n\n\n@sample samples.collections.Collections.Filtering.filterIndexed\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UShortArray.filterIndexedTo(predicate: (index: Int, UShort) -> Boolean): List<UInt> {\n\n    return filterIndexedTo(ArrayList<UInt>(), predicate)\n\n    // Adds all elements matching the given [predicate] to the given [destination].\n    // @param [predicate] function that takes the index of an element and the element itself\n    // and returns the result of predicate evaluation on the element.\n    // \n    // @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun <C : MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {\n\n    forEachIndexed { index, element ->\n        if (predicate(index, element)) destination.add(element)\n    }\n\n    return destination\n}

\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {\n\n    forEachIndexed { index, element ->\n        if (predicate(index, element)) destination.add(element)\n    }\n\n    return destination\n}

\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {\n\n    forEachIndexed { index, element ->\n        if (predicate(index, element)) destination.add(element)\n    }\n\n    return destination\n}

\n\n/**\n * Appends all elements matching the given [predicate] to the given [destination].\n * @param [predicate] function that takes the index of an element and the element itself\n * and returns the result of predicate evaluation on the element.\n * \n * @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun \n
// Adds all elements matching the given [predicate] to the given [destination].\n// @param [predicate] function that takes the index of an element and the element itself\n// and returns the result of predicate evaluation on the element.\n// \n// @sample samples.collections.Collections.Filtering.filterIndexedTo\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {\n\n    return filterNotTo(ArrayList<UInt>(), predicate)\n\n    // Returns a list containing all elements not matching the given [predicate].\n    // \n    // @sample samples.collections.Collections.Filtering.filterNot\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {\n\n    return filterNotTo(ArrayList<ULong>(), predicate)\n\n    // Returns a list containing all elements not matching the given [predicate].\n    // \n    // @sample samples.collections.Collections.Filtering.filterNot\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {\n\n    return filterNotTo(ArrayList<UByte>(), predicate)\n\n    // Returns a list containing all elements not matching the given [predicate].\n    // \n    // @sample samples.collections.Collections.Filtering.filterNot\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {\n\n    return filterNotTo(ArrayList<UShort>(), predicate)\n\n    // Returns a list containing all elements not matching the given [predicate].\n    // \n    // @sample samples.collections.Collections.Filtering.filterNot\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun
UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {
    return filterNotToArrayList<UInt>(UShortArray()(predicate))
    \n    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterNot
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public fun UShortArray.filterNotTo(destination: Collection<UShort>, predicate: (UShort) -> Boolean): Collection<UShort> {
 *    return filterNotToArrayList<UInt>(UShortArray()(predicate))
 *    for (element in this) if (!predicate(element)) destination.add(element)
 *    return destination
 *}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterNot
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public fun UIntArray.filterNotTo(destination: Collection<UInt>, predicate: (UInt) -> Boolean): Collection<UInt> {
 *    for (element in this) if (!predicate(element)) destination.add(element)
 *    return destination
 *}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterNot
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public fun ULongArray.filterNotTo(destination: Collection<ULong>, predicate: (ULong) -> Boolean): Collection<ULong> {
 *    for (element in this) if (!predicate(element)) destination.add(element)
 *    return destination
 *}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterNot
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public fun UByteArray.filterNotTo(destination: Collection<UByte>, predicate: (UByte) -> Boolean): Collection<UByte> {
 *    for (element in this) if (!predicate(element)) destination.add(element)
 *    return destination
 *}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 * @sample samples.collections.Collections.Filtering.filterNot
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 * @kotlin.internal.InlineOnly
 * public fun UShortArray.filterNotTo(destination: Collection<UShort>, predicate: (UShort) -> Boolean): Collection<UShort> {
 *    for (element in this) if (!predicate(element)) destination.add(element)
 *    return destination
 *}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: IntRange): List<UByte> {
    if (!indices.isEmpty()) return listOff()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList(null)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.slice(indices: IntRange): List<UShort> {
    if (!indices.isEmpty()) return listOff()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList(null)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.slice(indices: Iterable<Int>): List<UInt> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UInt>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.slice(indices: Iterable<Int>): List<ULong> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<ULong>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.slice(indices: Iterable<Int>): List<UShort> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sliceArray(indices: Collection<Int>): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: Collection<Int>): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: Collection<Int>): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: Collection<Int>): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sliceArray(indices: IntRange): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: IntRange): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: IntRange): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: IntRange): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.take(n: Int): List<UInt> {

/*@

*/
require(n >= 0) { "Requested element count $n is less than zero." }
if (n == 0) return emptyList()
if (n >= size) return toList()
if (n == 1) return listOf(this[0])
if (n == size) return list
for (item in this) {
    list.add(item)
    if (++count == n)
        break
}
return list

/**
 * Returns a list containing first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.take(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.takeLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UShort>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun
Open Source Used In DNAC Platform Arica

```kotlin
UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList().toList()//

*i *samples.Collections.Transformations.take

*\@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun

UIntArray.reverse(): Unit {
    storage.reverse()
}
```

This page contains code snippets related to the `takeLastWhile` and `reverse` methods in the `kotlin.collections` package. The `takeLastWhile` method is used to return a list containing last elements satisfying a given predicate. The `reverse` method reverses the elements in an array in-place.

The code is written in Kotlin and demonstrates the usage of these methods with various array types such as `UIntArray`, `ULongArray`, `UByteArray`, `UShortArray`, `ULongArray`, and `UByteArray`. Each method is accompanied by a brief description of its functionality and usage examples.
or [toIndex] is greater than the size of this array.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

/*@ Since Kotlin("1.4")*/ n@ExperimentalUnsignedTypes n@kotlin.internal.InlineOnly
public inline fun UIntArray.reverse(fromIndex: Int, toIndex: Int): Unit {
    storage.reverse(fromIndex, toIndex)n
}Reverses elements of the array in the specified range in-place.
* @param fromIndex the start of the range (inclusive) to reverse.
* @param toIndex the end of the range (exclusive) to reverse.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws IllegalArgumentExce...
```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.shuffle(): Unit {
    shuffle(Random)
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Randomly shuffles elements in this array in-place using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sorted(): List<UInt> {
    return copyOf().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sorted(): List<ULong> {
    return copyOf().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sorted(): List<UByte> {
    return copyOf().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sorted(): List<UShort> {
    return copyOf().apply { sort() }.asList()
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortedArray(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

---

Open Source Used In DNAC Platform Arica 9717
/*
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sortedArray(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sortedArray(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sortedArrayDescending(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sortedArrayDescending(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.asByteArray(): ByteArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [IntArray], which is a view of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asIntArray(): IntArray {
    return storage
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UIntArray.asList(): List<UInt>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>
a signed reinterpretation\n* of the corresponding element of this array.\n
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ULongArray.asLongArray(): LongArray {\n    return storage\n}

/**
 * Returns an array of type [ShortArray], which is a view of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.asShortArray(): ShortArray {\n    return storage\n}

/**
 * Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation\n* of the corresponding element of this array.\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
ByteArray.asUByteArray(): UByteArray {
    return UByteArray(this)\n}

/**
 * Returns an array of type [UIntArray], which is a view of this array where each element is an unsigned reinterpretation\n* of the corresponding element of this array.\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
IntArray.asUIntArray(): UIntArray {
    return UIntArray(this)\n}

/**
 * Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation\n* of the corresponding element of this array.\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
LongArray.asULongArray(): ULongArray {\n    return storage\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun UIntArray.contentEquals(other: UIntArray): Boolean {\n    return this.contentEquals(other)\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun ULongArray.contentEquals(other: ULongArray): Boolean {\n    return this.contentEquals(other)\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun UByteArray.contentEquals(other: UByteArray): Boolean {\n    return this.contentEquals(other)\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun UShortArray.contentEquals(other: UShortArray): Boolean {\n    return this.contentEquals(other)\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun UIntArray?.contentEquals(other: UIntArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)\n}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n* i.e. contain the same number of the same elements in the same order.\n*\n@SinceKotlin("1.3")\n@DeprecatedSinceKotlin(hiddenSince = \"1.4\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun ULongArray?.contentEquals(other: ULongArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)\n}
equal to one another, i.e. contain the same number of the same elements in the same order.

@SinceKotlin("1.4")@ExperimentalUnsignedTypes@public infix fun UByteArray?.contentEquals(other: UByteArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)??true if the two specified arrays are structurally equal to one another, i.e. contain the same number of the same elements in the same order.

@SinceKotlin("1.4")@ExperimentalUnsignedTypes@public infix fun UShortArray?.contentEquals(other: UShortArray?): Boolean {
    return this?.storage.contentEquals(other?.storage)??true
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.4")@ExperimentalUnsignedTypes@public fun UIntArray.contentHashCode(): Int {
    return this.contentHashCode()??true
}

/**
 * Returns a hash code based on the contents of this array as if it is `List<*>`.
 */
@SinceKotlin("1.3")@DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UIntArray.contentHashCode(): Int {
    return this.contentHashCode()??true
}

/**
 * Returns a hash code based on the contents of this array as if it is `List<*>`.
 */
@SinceKotlin("1.3")@DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UIntArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()??true
}

/**
 * Returns a hash code based on the contents of this array as if it is `List<*>`.
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun ULongArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()??true
}

/**
 * Returns a hash code based on the contents of this array as if it is `List<*>`.
 */
@SinceKotlin("1.3")@DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UByteArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()??true
}

/**
 * Returns a hash code based on the contents of this array as if it is `List<*>`.
 */
@SinceKotlin("1.3")@DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UShortArray?.contentHashCode(): Int {
    return this?.storage.contentHashCode()??true
}

/**
 * Returns a string representation of the contents of the specified array as if it is `List<*>`.
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UByteArray?.contentToString(): String {
    return this.contentToString()??true
}

/**
 * Returns a string representation of the contents of the specified array as if it is `List<*>`.
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UShortArray?.contentToString(): String {
    return this.contentToString()??true
}

/**
 * Returns a string representation of the contents of the specified array as if it is `List<*>`.
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UIntArray.contentToString(): String {
    return this.contentToString()??true
}

/**
 * Returns a string representation of the contents of the specified array as if it is `List<*>`.
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun UIntArray?.contentToString(): String {
    return this?.storage.contentToString()??true
}

/**
 * Returns a string representation of the contents of the specified array as if it is `List<*>`.
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")@ DeprecatedSinceKotlin(hiddenSince = "1.4")@ExperimentalUnsignedTypes@public fun ULongArray?.contentToString(): String {
    return this?.storage.contentToString()??true
}
Kotlin compiler 1.4 to avoid deprecation
warning.

@SinceKotlin("1.3")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.contentToString(): String {
    return this.contentToString() ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

@SinceKotlin("1.4")
@DeprecatedSinceKotlin(hiddenSince = "1.4")
@ExperimentalUnsignedTypes
public fun UShortArray?.contentToString(): String {
    return this?.contentToString() ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray?.contentToString(): String {
    return this?.joinToString(", ", "[", "]") ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray?.contentToString(): String {
    return this?.joinToString(", ", "[", "]") ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray?.contentToString(): String {
    return this?.joinToString(", ", "[", "]") ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray?.contentToString(): String {
    return this?.joinToString(", ", "[", "]") ?: "null"
} * Returns a string representation of the contents of the specified array as if it is [List].

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * 
 * @param destination the array to copy to, 0 by default.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * 
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.
 * 
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UIntArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ULongArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UByteArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}
@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @returns the [destination] array.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @returns the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

* @returns the [destination] array.

* @returns the array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.

* @returns the [destination] array.
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOf(newSize: Int): UByteArray {
    return UByteArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param fromIndex the start of the range (inclusive) to copy.
 * @param toIndex the end of the range (exclusive) to copy.
 * @param element the value to fill the copy.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray {
    return ULongArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.fill(element: UInt, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toInt(), fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive) to fill, 0 by default.
 * @param toIndex the end of the range (exclusive) to fill, size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toLong(), fromIndex, toIndex)
}
Fills this array or its subrange with the specified [element] value. This method fills the range from [fromIndex] to [toIndex]. If [fromIndex] is greater than [toIndex], this method fills from [fromIndex] to the end of the array. If [fromIndex] is less than zero or [toIndex] is greater than the size of this array, this method throws an exception.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toByte(), fromIndex, toIndex)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toShort(), fromIndex, toIndex)
}

Returns the range of valid indices for the array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UIntArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val ULongArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UByteArray.indices: IntRange
    get() = storage.indices

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.indices: IntRange
    get() = storage.indices

Returns the last valid index for the array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UIntArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val ULongArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UByteArray.lastIndex: Int
    get() = storage.lastIndex

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.lastIndex: Int
    get() = storage.lastIndex

Returns an array containing all elements of the original array and then the given [element].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.plus(element: UInt): UIntArray {
    return UIntArray(storage + element.toInt())
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage + element.toLong())
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage + element.toByte())
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return UShortArray(storage + element.toShort())
}

Returns an array containing all elements of the original array and then all elements of the given [elements] collection.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return UIntArray(result)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return ULongArray(result)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
}
range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array
by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater
than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater
than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * @sample samplesollections.Arrays.Sorting.sortRangeOfArray

*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    sortArray(this,
fromIndex, toIndex)
}

/**
 * Sorts a range in the array in-place.
 * 
 * @param fromIndex the start of the range (inclusive) to sort, 0 by default.\n * @param toIndex the end of the range (exclusive) to sort, size of this array
by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater
than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n * 
 * @sample samplesollections.Arrays.Sorting.sortRangeOfArray

*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    sortArray(this,
fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.\n * 
 * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater
than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UIntArray.sortDescending(fromIndex: Int,
toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Sorts elements of the array in the specified range in-place.
 * The elements are sorted descending according to their natural sort order.\n * 
 * @param fromIndex the start of the range (inclusive) to sort.\n * @param toIndex the end of the range (exclusive) to sort.\n * 
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater
than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n
*/
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun ULongArray.sortDescending(fromIndex: Int,
toIndex: Int): Unit {
    sort(fromIndex, toIndex)
    reverse(fromIndex, toIndex)
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.toIntArray(): IntArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.toByteArray(): ULongArray.sortDescending(fromIndex: Int,
toIndex: Int): Unit {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.toByteArray(): UByteArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    return storage.copyOf()
}

/**
 * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.toIntArray(): UIntArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    return storage.copyOf()
}

/**
 * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.toByteArray(): UByteArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    return storage.copyOf()
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.toByteArray(): UByteArray.sort(fromIndex: Int = 0,
toIndex: Int = size): Unit {
    return storage.copyOf()
}
ULongArray.toLongArray(): LongArray {
    return storage.copyOf()
} /* Returns an array of type [ShortArray], which is a copy of this array where each element is a signed reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
} /* Returns a *typed* object array containing all of the elements of this primitive array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@public fun UIntArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] } /* Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
} /* Returns an array of UByteArray containing all of the elements of this generic array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@public fun UIntArray.toUIntArray(): UIntArray {
    return UIntArray(size) { index -> this[index] } /* Returns an array of type [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun ULongArray.toULongArray(): ULongArray {
    return ULongArray(size) { index -> this[index] } /* Returns an array of type [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun UShortArray.toUShortArray(): UShortArray {
    return UShortArray(this.copyOf())
} /* Returns an array of UShortArray containing all of the elements of this generic array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@public fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] } /* Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun IntArray.toIntArray(): IntArray {
    return IntArray(size) { index -> this[index] } /* Returns an array of type [IntArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun ShortArray.toShortArray(): ShortArray {
    return ShortArray(size) { index -> this[index] } /* Returns an array of type [ShortArray], which is a copy of this array where each element is an unsigned reinterpretation * of the corresponding element of this array. */

*U@SinceKotlin("1.3")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun ShortArray.toShortArray(): ShortArray {
    return ShortArray(this.copyOf())
} /* Returns a *Map* where keys are elements from the given array and values are *n* produced by the [valueSelector] function applied to each element. *n* *n* *n* If any two elements are equal, the last one gets added to the map. *n* *n* *n* The returned map preserves the entry iteration order of the original array. *n* *n* *n* *n* *n* *n* @sample samples.collections.Collections.Transformations.associateWith

*U@SinceKotlin("1.4")@n@ExperimentalUnsignedTypes@n@kotlin.internal.InlineOnly@n@public inline fun <V> UIntArray.associateWith(valueSelector: (UInt) -> V): Map<UInt, V> {
    return result = LinkedHashMap<UInt,
V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Returns a [Map] where keys are elements from the given array and values are
 * produced by the [valueSelector] function applied to each element.\n * * * * If any two elements are equal, the last one gets added to the map.\n * * * * The returned map preserves the entry iteration order of the original array.\n * * * * @sample samples.collections.Collections.Transformations.associateWith
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V> ULongArray.associateWith(valueSelector: (ULong) -> V): Map<ULong, V> {
    val result =
    LinkedHashMap<ULong, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Returns a [Map] where keys are elements from the given array and values are
 * produced by the [valueSelector] function applied to each element.\n * * * * If any two elements are equal, the last one gets added to the map.\n * * * * The returned map preserves the entry iteration order of the original array.\n * * * * @sample samples.collections.Collections.Transformations.associateWith
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V> UByteArray.associateWith(valueSelector: (UByte) -> V): Map<UByte, V> {
    val result =
    LinkedHashMap<UByte, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Returns a [Map] where keys are elements from the given array and values are
 * produced by the [valueSelector] function applied to each element.\n * * * * If any two elements are equal, the last one gets added to the map.\n * * * * The returned map preserves the entry iteration order of the original array.\n * * * * @sample samples.collections.Collections.Transformations.associateWith
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V> UShortArray.associateWith(valueSelector: (UShort) -> V): Map<UShort, V> {
    val result =
    LinkedHashMap<UShort, V>(mapCapacity(size).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each
 * element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function
 * applied to that key.\n * * * * If any two elements are equal, the last one overwrites the former value in the map.\n * * * * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UInt, in V>> UIntArray.associateWithTo(destination: M, valueSelector: (UInt) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * * * * If any two elements are equal, the last one overwrites the former value in the map.\n * * * * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in ULong, in V>> ULongArray.associateWithTo(destination: M, valueSelector: (ULong) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * * * * If any two elements are equal, the last one overwrites the former value in the map.\n * * * * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M : MutableMap<in UByte, in V>> UByteArray.associateWithTo(destination: M, valueSelector: (UByte) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given array,\n * where key is the element itself and value is provided by the [valueSelector] function applied to that key.\n * * * * If any two elements are equal, the last one overwrites the former value in the map.\n * * * * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, M :
Open Source Used In DNAC Platform Arica 9730

```kotlin
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UIntArray.flatMapIndexedTo(destination: C, transform: (Int, UInt) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of
 * and its index in the original array, to the given [destination].
 */
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapIndexedTo(destination: C, transform: (Int, ULong) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element
 * and its index in the original array, to the given [destination].
 */
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapIndexedTo(destination: C, transform: (Int, UByte) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element
 * and its index in the original array, to the given [destination].
 */
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapIndexedTo(destination: C, transform: (Int, UShort) -> Iterable<R>): C {
    var index = 0
    for (element in this) {
        val list = transform(index++, element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector]
 * function
 */
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : Map<in Short, in List<R>>> UShortArray.flatMapTo(destination: C, keySelector: (UShort) -> Short): C {
    for (element in this) {
        val key = keySelector(element)
        val list = transform(key, element)
        destination.add(list)
    }
    return destination
}
```

*Groups elements of the original array by the key returned by the given [keySelector] function* applied to each element and returns a map where each group key is associated with a list of
The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.

* \n
* The returned map preserves the entry iteration order of the keys produced from the original array.
returned by the \[\text{valueTransform}\] function applied to each element of the original array
* by the key returned by the given \[\text{keySelector}\] function applied to the element
* and returns a map where each key is associated with a list of corresponding values.
* The returned map preserves the entry iteration order of the keys produced from the original array.
* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

*\[@\text{Since}\text{Kotlin}\("1.3\)\]*\text{\@ExperimentalUnsignedTypes}@\text{@kotlin.internal.InlineOnly}\text{@public inline fun <K, V> UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> {\}} \text{return groupByKeyTo(LinkedListHashMap<K, MutableList<V>>(), keySelector, valueTransform)}\text{\n\n\n**Groups elements of the original array by the key returned by the given \[keySelector\] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n* \n\n\n* @sample samples.collections.Collections.Transformations.groupBy

*\[@\text{Since}\text{Kotlin}\("1.3\)\]*\text{\@ExperimentalUnsignedTypes}@\text{@kotlin.internal.InlineOnly}\text{@public inline fun <K, M : MutableMap<K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M {\}} \text{\n\n\n\nfor (element in this) {\n    val key = keySelector(element)\n    val list = destination.getOrElse(key) {\n        ArrayList<V>()\n    }\n    list.add(valueTransform(element))\n}\n\n\n**Groups elements of the original array by the key returned by the given \[keySelector\] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n* \n\n\n* @sample samples.collections.Collections.Transformations.groupBy
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K, valueTransform: (ULong) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K, valueTransform: (UShort) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.map(transform: (UInt) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.map(transform: (ULong) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.mapIndexed(transform: (index: Int, UShort) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
that takes the index of an element and the element itself as input and returns the result of the transform applied to the element.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UByteArray.mapIndexed(transform: (Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the
 * given [transform] function to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself as
 * input and returns the result of the transform applied to the element.
 * @param [target] the destination for the results.
 * @return a list of results.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UShortArray.mapIndexed(transform: (Int, UShort) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}
```

```kotlin
/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself as
 * input and returns the result of the transform applied to the element.
 * @param [destination] the destination for the results.
 * @return the updated destination.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : Collection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (Int, UInt) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```

```kotlin
/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself as
 * input and returns the result of the transform applied to the element.
 * @param [destination] the destination for the results.
 * @return the updated destination.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : Collection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (Int, ULong) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself as
 * input and returns the result of the transform applied to the element.
 * @param [destination] the destination for the results.
 * @return the updated destination.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : Collection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (Int, UByte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

```kotlin
/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself as
 * input and returns the result of the transform applied to the element.
 * @param [destination] the destination for the results.
 * @return the updated destination.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : Collection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (Int, UShort) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```
MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. @sample @SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UIntArray.withIndex():
 * Iterable<IndexedValue<UInt>> { return IndexingIterable { iterator() } }
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun ULongArray.withIndex():
    return IndexingIterable { iterator() }

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. @sample @SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UByteArray.withIndex():
 * Iterable<IndexedValue<UByte>> { return IndexingIterable { iterator() } }
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@public fun UShortArray.withIndex():
    return IndexingIterable { iterator() }

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UIntArray.all(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UIntArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    ULongArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UByteArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UShortArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UIntArray.any(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    ULongArray.any(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UByteArray.any(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly@public inline fun
    UShortArray.any(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
ULongArray.any(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
} * Returns 'true' if at least one element matches the given [predicate].

@sample samples.collections.Collections.Aggregates.anyWithPredicate

UByteArray.any(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
} * Returns 'true' if at least one element matches the given [predicate].

@sample samples.collections.Collections.Aggregates.anyWithPredicate

UCharArray.any(predicate: (UChar) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
} * Returns 'true' if at least one element matches the given [predicate].

@sample samples.collections.Collections.Aggregates.anyWithPredicate

UIntArray.count(predicate: (UInt) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
} * Returns the number of elements matching the given [predicate].

@sample samples.collections.Collections.Aggregates.countWithPredicate

UCharArray.count(predicate: (UChar) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
} * Returns the number of elements matching the given [predicate].

@sample samples.collections.Collections.Aggregates.countWithPredicate

UShortArray.count(predicate: (UShort) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
} * Returns the number of elements matching the given [predicate].

@sample samples.collections.Collections.Aggregates.countWithPredicate

UByteArray.count(predicate: (UByte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
} * Returns the number of elements matching the given [predicate].

@sample samples.collections.Collections.Aggregates.countWithPredicate

    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} * Accumulates value starting with [initial] value and applying [operation] from left to right
    * to current accumulator value and each element.

@sample samples.collections.Collections.Aggregates.foldWithOperation

UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
} * Accumulates value starting with [initial] value and applying [operation] from left to right
    * to current accumulator value and each element.

@sample samples.collections.Collections.Aggregates.foldWithOperation
* @param [operation] function that takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element) return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from left to right. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element) return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from left to right. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element) return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from left to right. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element) return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from left to right. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from right to left. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from right to left. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

* Accumulates value starting with [initial] value and applying [operation] from right to left. Returns the specified [initial] value if the array is empty. Returns the specified [initial] value when the array is empty, and the element itself, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```
to left\n* to each element and current accumulator value.\n * \n * Returns the specified [initial] value if the array is empty.\n * \n * @param [operation] function that takes an element and current accumulator value, and calculates the next accumulator value.\n
```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.foldRightIndexed(initial: R, operation: (UShort, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)\n    }\n    return accumulator\n}
```
Performs the given [action] on each element, providing sequential index with the element.

```kotlin
public inline fun UIntArray.forEachIndexed(action: (index: Int, UInt) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with
 * the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.forEachIndexed(action: (index: Int, ULong) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with
 * the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.forEachIndexed(action: (index: Int, UByte) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with
 * the element.
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.forEachIndexed(action: (index: Int, UShort) -> Unit): Unit {
    var index = 0
    for (item in this)
        action(index++, item)
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.max(): UInt? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.max(): ULong? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.max(): UByte? {
    return maxOrNull()
}

@Deprecated("Use maxOrNull instead.",
ReplaceWith("maxOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.max(): UShort? {
    return maxOrNull()
}

@Deprecated("Use maxByOrNull instead.",
ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxBy(selector: (UInt) -> R): UInt? {
    return maxByOrNull(selector)
}

@Deprecated("Use maxByOrNull instead.",
ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxBy(selector: (ULong) -> R): ULong? {
    return maxByOrNull(selector)
}

@Deprecated("Use maxByOrNull instead.",
ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxBy(selector: (UByte) -> R): UByte? {
    return maxByOrNull(selector)
}

@Deprecated("Use maxByOrNull instead.",
ReplaceWith("maxByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {
    return maxByOrNull(selector)
}
```
`null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
@ExperimentalUnsignedTypes
public inline fun <R : Comparable<R>> UIntArray.maxByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        var e = this[i]
        var v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
```

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.maxByOrNull(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        var e = this[i]
        var v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
```

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        var e = this[i]
        var v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
```

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.maxOf(selector: (UInt) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.maxOf(selector: (ULong) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.maxOf(selector: (UByte) -> Double): Double {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        var v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
```
maxOf(maxValue, v) in return maxValue
} /* * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.maxOf(selector: (UShort) -> Double): Double { if (isEmpty()) throw NoSuchElementException()

var maxValue = selector(this[0])

for (i in 1..lastIndex) {
    val v = selector(this[i])

    if (maxValue < v) {
        maxValue = v
    }
}

return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.maxOf(selector: (UInt) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()

    var maxValue = selector(this[0])

    for (i in 1..lastIndex) {
        val v = selector(this[i])

        maxValue = maxOf(maxValue, v)
    }

    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.maxOf(selector: (ULong) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()

    var maxValue = selector(this[0])

    for (i in 1..lastIndex) {
        val v = selector(this[i])

        maxValue = maxOf(maxValue, v)
    }

    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.maxOf(selector: (UByte) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()

    var maxValue = selector(this[0])

    for (i in 1..lastIndex) {
        val v = selector(this[i])

        maxValue = maxOf(maxValue, v)
    }

    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.maxOf(selector: (UInt) -> R): R { if (isEmpty()) throw NoSuchElementException()

var maxValue = selector(this[0])

for (i in 1..lastIndex) {
    val v = selector(this[i])

    if (maxValue < v) {
        maxValue = v
    }
}

return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOf(selector: (UShort) -> R): R { if (isEmpty()) throw NoSuchElementException()

var maxValue = selector(this[0])

for (i in 1..lastIndex) {
    val v = selector(this[i])

    if (maxValue < v) {
        maxValue = v
    }
}

return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.maxOf(selector: (UByte) -> R): R { if (isEmpty()) throw NoSuchElementException()

var maxValue = selector(this[0])

for (i in 1..lastIndex) {
    val v = selector(this[i])

    if (maxValue < v) {
        maxValue = v
    }
}

return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function \n * applied to each element in the array. In * if any of values produced by [selector] function is NaN, the returned result is NaN. * @throws NoSuchElementException if the array is empty. */

@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.maxOf(selector: (UShort) -> R): R { if (isEmpty()) throw NoSuchElementException()
Comparable<R>> ULongArray.maxOf(selector: (ULong) -> R): R {
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxVal < v) {
            maxVal = v
        }
    }
    return maxVal
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array.
 */

Comparable<R>> UByteArray.maxOf(selector: (UByte) -> R): R {
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxVal < v) {
            maxVal = v
        }
    }
    return maxVal
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

Comparable<R>> UShortArray.maxOf(selector: (UShort) -> R): R {
    var maxVal = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxVal < v) {
            maxVal = v
        }
    }
    return maxVal
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

Comparable<R>> UIntArray.maxOfOrNull(selector: (UInt) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

Comparable<R>> ULongArray.maxOfOrNull(selector: (ULong) -> R): Double? {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */

Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> Double): Double? {
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}
public inline fun UIntArray.maxOfOrNull(selector: (UInt) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun ULongArray.maxOfOrNull(selector: (ULong) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun UByteArray.maxOfOrNull(selector: (UByte) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun UShortArray.maxOfOrNull(selector: (UShort) -> Float): Float? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        maxValue = maxOf(maxValue, v)
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> UIntArray.maxOfOrNull(selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> ULongArray.maxOfOrNull(selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> UByteArray.maxOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> UShortArray.maxOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> Comparable<*>.maxOfOrNull(selector: (Comparable<*>?) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}

/**
 * Returns the largest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
public inline fun <R : Comparable<R>> Comparable<*>.maxOfOrNull(selector: (Comparable<*>?) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (maxValue < v) {
            maxValue = v
        }
    }
    return maxValue
}
maxValue = selector(this[0])
for (i in 1..lastIndex) {
    val v = selector(this[i])
    if (maxValue < v) {
        maxValue = v
    }
}
return maxValue

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
* @throws NoSuchElementException if the array is empty.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.maxOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
* @throws NoSuchElementException if the array is empty.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.maxOfWith(comparator: Comparator<in R>, selector: (ULong) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
* @throws NoSuchElementException if the array is empty.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.maxOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
* @throws NoSuchElementException if the array is empty.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.maxOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}

/**
* Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
*/
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
function applied to each element in the array or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UByte) -> R): R? {
    if (isEmpty())
        return null
    var maxValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(maxValue, v) < 0) {
            maxValue = v
        }
    }
    return maxValue
}
```

Returns the largest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.maxOfWithOrNull(comparator: Comparator<in R>, selector: (UShort) -> R): R? {
    if (isEmpty())
        return null
    var max
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(max, v) < 0) {
            max = v
        }
    }
    return max
}
```

Returns the largest element or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.maxOrNull(): UInt? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}
```

Returns the largest element or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.maxOrNull(): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}
```

Returns the largest element or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.maxOrNull(): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}
```

Returns the largest element or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.maxWith(comparator: Comparator<in UInt>): UInt? {
    return maxWithOrNull(comparator)
}
```

Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {
    return maxWithOrNull(comparator)
}
```

Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {
    return maxWithOrNull(comparator)
}
```

Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
first element having the largest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UByteArray.maxWithOrNull(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public fun UShortArray.maxWithOrNull(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.min(): UInt? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.min(): ULong? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.min(): UByte? {
    return minOrNull()
}

@Deprecated("Use minOrNull instead.",
ReplaceWith("minOrNull()"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.min(): UShort? {
    return minOrNull()
}

@Deprecated("Use minByOrNull instead.",
ReplaceWith("minByOrNull(selector)"))
@DeprecatedSinceKotlin(warningSince = "1.4")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.minByOrNull(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.minByOrNull(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.minByOrNull(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minByOrNull(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}
```
selector(minElem) for (i in 1..lastIndex) {\n    val e = this[i]\n    val v = selector(e)\n    if (minValue > v) \n    {\n        minElem = e\n        minValue = v\n    }\n}\n    return minElem\n}\n
/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * @throws NoSuchElementException if the array is empty.
 * 
 * @param selector the function used to select a value from each element.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <T : Comparable<T>> Array<T>.minByOrNull(selector: (T) -> Double): T? {
    if (isEmpty()) throw NoSuchElementException()\n    var minValue = selector(this[0])\n    for (i in 1..lastIndex) {\n        val v = selector(this[i])\n        minValue = minOf(minValue, v)\n    }\n    return minValue
}
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.minOf(selector: (UInt) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.minOf(selector: (ULong) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.minOf(selector: (UByte) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.minOf(selector: (UShort) -> Float): Float {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced
 * by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.minOf(selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.minOf(selector: (ULong) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.minOf(selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * 
 * If any of values produced by [selector] function is `NaN`,
 * the returned result is `NaN`. 
 * @throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType

@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOf(selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOf(selector: (UShort) -> R): R {
    if (isEmpty()) throw
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or null if there are no elements.
 * If any of values produced by [selector] function is 'NaN', the returned result
 * is 'NaN'.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULLongArray.minOfOrNull(selector: (ULLong) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or null if there are no elements.
 * If any of values produced by [selector] function is 'NaN', the returned result
 * is 'NaN'.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.minOfOrNull(selector: (UByte) -> Double): Double? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or null if there are no elements.
 * If any of values produced by [selector] function is 'NaN', the returned result
 * is 'NaN'.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.minOfOrNull(selector: (UInt) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}

/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or null if there are no elements.
 * If any of values produced by [selector] function is 'NaN', the returned result
 * is 'NaN'.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
ByLambdaReturnType
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.minOfOrNull(selector: (ULong) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
applied to each element in the array or `null` if there are no elements. If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.

```kotlin
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.minOfOrNull(selector: (UByte) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 * If any of values produced by [selector] function is `NaN`, the returned result is `NaN`.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.minOfOrNull(selector: (UShort) -> Float): Float? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        minValue = minOf(minValue, v)
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UIntArray.minOfOrNull(selector: (UInt) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> ULongArray.minOfOrNull(selector: (ULong) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UByteArray.minOfOrNull(selector: (UByte) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function
 * applied to each element in the array or `null` if there are no elements.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparable<R>> UShortArray.minOfOrNull(selector: (UShort) -> R): R? {
    if (isEmpty()) return null
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (minValue > v) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value according to the provided [comparator] among all values produced by [selector] function applied to each element in the array.
 * `throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>, K : Comparable<K>> UIntArray.minOfWith(comparator: Comparator<in R>, selector: (UInt) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * `throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UByteArray.minOfWith(comparator: Comparator<in R>, selector: (UByte) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```

```kotlin
/**
 * Returns the smallest value among all values produced by [selector] function applied to each element in the array.
 * `throws NoSuchElementException if the array is empty.
 */
@SinceKotlin("1.4")
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@LambdaReturnType
@OptIn(kotlin.experimental.ExperimentalTypeInference::class)
@OverloadResolution
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R : Comparator<in R>> UShortArray.minOfWith(comparator: Comparator<in R>, selector: (UShort) -> R): R {
    if (isEmpty()) throw NoSuchElementException()
    var minValue = selector(this[0])
    for (i in 1..lastIndex) {
        val v = selector(this[i])
        if (comparator.compare(minValue, v) > 0) {
            minValue = v
        }
    }
    return minValue
}
```
*\an@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\nULongArray.minOfWith\ncomparator: Comparator\n{in R}, selector: (ULong) -> R: R
\n\nvar minValue = selector(this[0])
for (i in 1..lastIndex)
var v = selector(this[i])
if (comparator.compare(minValue, v) > 0)
var minValue = v
\n\n**\n\n* Returns the smallest value according to the provided [comparator]\n* among all values produced by [selector] function applied to each element in the array.
\n* \n* \n* \n@throws NoSuchElementException if the array is empty.

*\an@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\nUByteArray.minOfWith\ncomparator: Comparator\n{in R}, selector: (UByte) -> R: R
\n\nvar minValue = selector(this[0])
for (i in 1..lastIndex)
var v = selector(this[i])
if (comparator.compare(minValue, v) > 0)
var minValue = v
\n\n**\n\n* Returns the smallest value according to the provided [comparator]\n* among all values produced by [selector] function applied to each element in the array.
\n* \n* \n* \n@throws NoSuchElementException if the array is empty.

*\an@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\nUShortArray.minOfWith\ncomparator: Comparator\n{in R}, selector: (UShort) -> R: R
\n\nvar minValue = selector(this[0])
for (i in 1..lastIndex)
var v = selector(this[i])
if (comparator.compare(minValue, v) > 0)
var minValue = v
\n\n**\n\n* Returns the smallest value according to the provided [comparator]\n* among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\an@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\nUIntArray.minOfWithOrNullable\ncomparator: Comparator\n{in R}, selector: (UInt) -> R: R?\n\n\nvar minValue = selector(this[0])
for (i in 1..lastIndex)
var v = selector(this[i])
if (comparator.compare(minValue, v) > 0)
var minValue = v
\n\n**\n\n* Returns the smallest value according to the provided [comparator]\n* among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.

*\an@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\n\nUByteArray.minOfWithOrNullable\ncomparator: Comparator\n{in R}, selector: (UByte) -> R: R?\n\n\nvar minValue = selector(this[0])
for (i in 1..lastIndex)
var v = selector(this[i])
if (comparator.compare(minValue, v) > 0)
var minValue = v
\n\n**\n\n* Returns the smallest value according to the provided [comparator]\n* among all values produced by [selector] function applied to each element in the array or `null` if there are no elements.
Open Source Used In DNAC Platform Arica

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.none(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.none(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.none(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.none(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given [action] on each element and returns the array itself afterwards.
 * 
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.onEachIndexed(action: (index: Int, UInt) -> Unit): UIntArray {
    return apply {
        forEachIndexed(action)
    }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,
 * and returns the array itself afterwards.
 * 
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.onEachIndexed(action: (index: Int, ULong) -> Unit): ULongArray {
    return apply {
        forEachIndexed(action)
    }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,
 * and returns the array itself afterwards.
 * 
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.onEachIndexed(action: (index: Int, UByte) -> Unit): UByteArray {
    return apply {
        forEachIndexed(action)
    }
}

/**
 * Performs the given [action] on each element, providing sequential index with the element,
 * and returns the array itself afterwards.
 * 
 * @param [action] function that takes the index of an element and the element itself
 * and performs the action on the element.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.onEachIndexed(action: (index: Int, UShort) -> Unit): UShortArray {
    return apply {
        forEachIndexed(action)
    }
}
```
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.onEachIndexed(action: (index: Int, UByte) -> Unit): UByteArray {
    return apply {
        forEachIndexed(action) }
}

/**
 * Performs the given [action] on each element, providing sequential index
 * with the element,\n * and returns the array itself afterwards.\n * @param [action] function that takes the index of an element and the element itself\n * and performs the action on the element.\n */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.onEachIndexed(action: (index: Int, UShort) -> Unit): UShortArray {
    return apply {
        forEachIndexed(action) }
}

/**
 * Accumulates value starting with the first element and applying [operation]
 * from left to right\n * to current accumulator value and each element.\n * Throws an exception if this array is
 * empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * and calculates the next accumulator value.\n * @param [operation] function that takes current accumulator value and an element,\n * and calculates the next accumulator value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.\n"
            var accumulator = this[0] for (index in 1..lastIndex) {\n                accumulator = operation(accumulator, this[index])\n            }\n            return accumulator\n        )\n        return accumulator\n    var accumulator = this[0] for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}

/**
 * Accumulates value starting with the first element and applying [operation]
 * from left to right\n * to current accumulator value and each element.\n * Throws an exception if this array is
 * empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n * and calculates the next accumulator value.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.\n"
            var accumulator = this[0] for (index in 1..lastIndex) {\n                accumulator = operation(accumulator, this[index])\n            }\n            return accumulator\n        )\n        return accumulator\n    var accumulator = this[0] for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}

/**
 * Accumulates value starting with the first element and applying [operation]
 * from left to right\n * to current accumulator value and each element.\n * Throws an exception if this array is
 * empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.\n"
            var accumulator = this[0] for (index in 1..lastIndex) {\n                accumulator = operation(accumulator, this[index])\n            }\n            return accumulator\n        )\n        return accumulator\n    var accumulator = this[0] for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}

/**
 * Accumulates value starting with the first element and applying [operation]
 * from left to right\n * to current accumulator value and each element.\n * Throws an exception if this array is
 * empty. If the array can be empty in an expected way,\n * please use [reduceOrNull] instead. It returns `null` when its receiver is empty.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.\n"
            var accumulator = this[0] for (index in 1..lastIndex) {\n                accumulator = operation(accumulator, this[index])\n            }\n            return accumulator\n        )\n        return accumulator\n    var accumulator = this[0] for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}
samples.collections.Collections.Aggregates.reduce

*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Throws an exception if this array is empty. If the array can be empty in an expected way,
// please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
// @param [operation] function that takes the index of an element, current accumulator value and
// the element itself, and calculates the next accumulator value.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Throws an exception if this array is empty. If the array can be empty in an expected way,
// please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Throws an exception if this array is empty. If the array can be empty in an expected way,
// please use [reduceIndexedOrNull] instead. It returns `null` when its receiver is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Returns `null` if the array is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
UIntArray.reduceIndexedOrNull(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Returns `null` if the array is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
ULongArray.reduceIndexedOrNull(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Returns `null` if the array is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
UByteArray.reduceIndexedOrNull(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

// Accumulates value starting with the first element and applying [operation] from left to right to
// current accumulator value and each element with its index in the original array.
// Returns `null` if the array is empty.
// * and calculates the next accumulator value.

*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public inline fun ULongArray.reduceIndexedOrNull(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public inline fun UByteArray.reduceIndexedOrNull(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public inline fun UIntArray.reduceOrNull(operation: (acc: UInt, UInt) -> UInt): UInt? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public inline fun ULongArray.reduceOrNull(operation: (acc: ULong, ULong) -> ULong): ULong? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.reduceOrNull
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
public inline fun UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)
        accumulator = operation(accumulator, this[index])
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightIndexed(operation: (UInt, index: Int, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), index, accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexed(operation: (ULong, index: Int, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), index, accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexed(operation: (UByte, index: Int, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), index, accumulator)
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to left
 * to each element and current accumulator value.
 * Throws an exception if this array is empty. If the array can be empty in an expected way,
 * please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.
 * Calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0)
        accumulator = operation(get(index--), index, accumulator)
    return accumulator
}
next accumulator value

*\n* \n* @sample samples.collections.Collections Aggregates.reduceRight

*\n* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* @fun UIntArray.reduceRightIndexed\n
*operation: (index: Int, UInt, acc: UInt) -> UInt: UInt \n* var index = lastIndex\n
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\n* var accumulator = \n
* get(index--) \n
* while (index >= 0) \n
* { \n
* accumulator = operation(index, get(index), accumulator) \n
* --index \n
* } \n
* return accumulator \n
*\n*\n*\n*\n* Accumulates value starting with the last element and applying [operation]

* from right to left to each element with its index in the original array and current accumulator value.\n*\n*\n*\n*\n*\n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n*\n* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n*\n*\n*\n*\n* @param [operation] function that takes the index of an element, the element itself and current accumulator value.\n*\n* and calculates the next accumulator value.\n*\n* @sample samples.collections.Collections Aggregates.reduceRight

*\n* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* @fun ULongArray.reduceRightIndexed\n
*operation: (index: Int, ULong, acc: ULong) -> ULong: ULong \n* var index = lastIndex\n
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\n* var accumulator = \n
* get(index--) \n
* while (index >= 0) \n
* { \n
* accumulator = operation(index, get(index), accumulator) \n
* --index \n
* } \n
* return accumulator \n
*\n*\n*\n*\n*\n* Accumulates value starting with the last element and applying [operation]

* from right to left to each element with its index in the original array and current accumulator value.\n*\n*\n*\n*\n*\n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n*\n* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n*\n*\n*\n*\n* @param [operation] function that takes the index of an element, the element itself and current accumulator value.\n*\n* and calculates the next accumulator value.\n*\n* @sample samples.collections.Collections Aggregates.reduceRight

*\n* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* @fun UByteArray.reduceRightIndexed\n
*operation: (index: Int, UByte, acc: UByte) -> UByte: UByte \n* var index = lastIndex\n
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\n* var accumulator = \n
* get(index--) \n
* while (index >= 0) \n
* { \n
* accumulator = operation(index, get(index), accumulator) \n
* --index \n
* } \n
* return accumulator \n
*\n*\n*\n*\n*\n* Accumulates value starting with the last element and applying [operation]

* from right to left to each element with its index in the original array and current accumulator value.\n*\n*\n*\n*\n*\n* Throws an exception if this array is empty. If the array can be empty in an expected way,\n*\n* please use [reduceRightIndexedOrNull] instead. It returns `null` when its receiver is empty.\n*\n*\n*\n*\n* @param [operation] function that takes the index of an element, the element itself and current accumulator value.\n*\n* and calculates the next accumulator value.\n*\n* @sample samples.collections.Collections Aggregates.reduceRight

*\n* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* @fun UShortArray.reduceRightIndexed\n
*operation: (index: Int, UShort, acc: UShort) -> UShort: UShort \n* var index = lastIndex\n
* if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.\n* var accumulator = \n
* get(index--) \n
* while (index >= 0) \n
* { \n
* accumulator = operation(index, get(index), accumulator) \n
* --index \n
* } \n
* return accumulator \n
*\n*\n*\n*\n*\n* Accumulates value starting with the last element and applying [operation]

* from right to left to each element with its index in the original array and current accumulator value.\n*\n*\n*\n*\n*\n* Returns `null` if the array is empty.\n*\n*\n*\n*\n* @param [operation] function that takes the index of an element, the element itself and current accumulator value.\n*\n* and calculates the next accumulator value.\n*\n* @sample samples.collections.Collections Aggregates.reduceRightOrNull

*\n* @SinceKotlin("1.4")@ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* @fun UIntArray.reduceRightIndexedOrNull\n
*operation: (index: Int, UInt, acc: UInt) -> UInt: UInt? \n* var index = lastIndex\n
* if (index < 0) return null \n* var accumulator = \n
* get(index--) \n
* while (index >= 0) \n
* { \n
* accumulator = operation(index, get(index), accumulator) \n
* --index \n
* } \n
* return accumulator \n
*\n*\n*\n*\n*\n* Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n*\n*\n*\n*\n*\n* Returns `null` if the array is empty.\n*\n*\n*\n*\n*
@param [operation] function that takes the index of an element, the element itself and current accumulator value.

* and calculates the next accumulator value.

```kotlin
public inline fun ULongArray.reduceRightIndexedOrNull(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
            --index
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. Returns `null` if the array is empty. * and calculates the next accumulator value.

```kotlin
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexedOrNull(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
            --index
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element and current accumulator value. Returns `null` if the array is empty. * and calculates the next accumulator value.

```kotlin
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
            --index
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element and current accumulator value. Returns `null` if the array is empty. * and calculates the next accumulator value.

```kotlin
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightIndexedOrNull(operation: (UInt, acc: UInt) -> UInt): UInt? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element and current accumulator value. Returns `null` if the array is empty. * and calculates the next accumulator value.

```kotlin
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexedOrNull(operation: (ULong, acc: ULong) -> ULong): ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element and current accumulator value. Returns `null` if the array is empty. * and calculates the next accumulator value.

```kotlin
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexedOrNull(operation: (UByte, acc: UByte) -> UByte): UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
```
while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with the last element and applying [operation] from right to
 * left to each element and current accumulator value.
 * Returns `null` if the array is empty.
 * @param [operation] function that takes an element and current accumulator value and calculates the next accumulator value.
 * @param index value.
 * @param accumulator initial value.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@kotlin.internal.InlineOnly
@ExperimentalUnsignedTypes
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param initial value.
 * @param accumulator initial value.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.runningFold(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param index value.
 * @param accumulator initial value.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.runningFold(initial: R, operation: (acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @param index value.
 * @param accumulator initial value.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.runningFold(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
(element in this) {\n    accumulator = operation(accumulator, element)\n    result.add(accumulator)\n}\nreturn result\n\n/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.\n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UIntArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}\n
/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.\n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> ULongArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}\n
/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.\n * Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.\n * @sample samples.collections.Collections.Aggregates.runningFold\n */\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UByteArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}\n
/**\n * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.\n * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.\n * @sample samples.collections.Collections.Aggregates.runningReduce\n */\n@SinceKotlin("1.4")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R> UShortArray.runningFoldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): List<R> {\n    if (isEmpty()) return listOf(initial)\n    val result = ArrayList<R>(size + 1).apply { add(initial) }\n    var accumulator = initial\n    for (index in indices) {\n        accumulator = operation(index, accumulator, this[index])\n        result.add(accumulator)\n    }\n    return result\n}
UIntArray.runningReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return
    val result = ArrayList<UInt>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts
 * with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.runningReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts
 * with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.runningReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that starts
 * with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
public inline fun UIntArray.runningReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original array and
 * current accumulator value that starts with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.runningReduce
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
itself, and calculates the next accumulator value:

```kotlin
*\n*\n* @sample samples.collections.Collections.Aggregates.runningReduce

*\n*\n* @SinceKotlin("1.4\")\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun ULongArray.runningReduceIndexed(operation: (Int, acc: ULong, ULong) -> ULong): List<ULong> {
* \n* if (isEmpty()) return emptyList()\n* \n* var accumulator = this[0]\n* \n* val result = ArrayList<ULong>(size).apply {
* \n*\n* for (index in 1 until size) {\n* \n* \n* accumulator = operation(index, accumulator, this[index])\n* \n*\n* result.add(accumulator)\n* \n*}\n* \n* return result()\n* \n*}\n* \n*\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n* \n* * Note that `acc` value passed to [operation] function should not be mutated;\n* * otherwise it would affect the previous value in resulting list.\n* \n* * @param [operation] function that takes the index of an element, current accumulator value\n* \n* * and the element itself, and calculates the next accumulator value.\n* \n* * @sample samples.collections.Collections.Aggregates.runningReduce

*\n*\n* @SinceKotlin("1.4")\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun UByteArray.runningReduceIndexed(operation: (Int, acc: UByte, UByte) -> UByte): List<UByte> {
* \n* if (isEmpty()) return emptyList()\n* \n* var accumulator = this[0]\n* \n* val result = ArrayList<UByte>(size).apply {
* \n*\n* for (index in 1 until size) {\n* \n* \n* accumulator = operation(index, accumulator, this[index])\n*\n* result.add(accumulator)\n* \n*}\n* \n* return result()\n* \n*}\n* \n*\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n* \n* * Note that `acc` value passed to [operation] function should not be mutated;\n* * otherwise it would affect the previous value in resulting list.\n* \n* * @param [operation] function that takes the index of an element, current accumulator value\n* \n* * and the element itself, and calculates the next accumulator value.\n* \n* * @sample samples.collections.Collections.Aggregates.runningReduce

*\n*\n* @SinceKotlin("1.4")\n* @ExperimentalUnsignedTypes\n* @kotlin.internal.InlineOnly\n* public inline fun UShortArray.runningReduceIndexed(operation: (Int, acc: UShort, UShort) -> UShort): List<UShort> {
* \n* if (isEmpty()) return emptyList()\n* \n* var accumulator = this[0]\n* \n* val result = ArrayList<UShort>(size).apply {
* \n*\n* for (index in 1 until size) {\n* \n* \n* accumulator = operation(index, accumulator, this[index])\n* \n*\n* result.add(accumulator)\n* \n*}\n* \n* return result()\n* \n*}\n* \n*\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* * to each element, its index in the original array and current accumulator value that starts with the first element of this array.\n* \n* * Note that `acc` value passed to [operation] function should not be mutated;\n* * otherwise it would affect the previous value in resulting list.\n* \n* * @param [operation] function that takes the index of an element, current accumulator value\n* \n* * and the element itself, and calculates the next accumulator value.\n* \n* * @sample samples.collections.Collections.Aggregates.runningReduce

*\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* @kotlin.internal.InlineOnly\n* public inline fun <R> UIntArray.scan(initial: R, operation: (acc: R, UInt) -> R): List<R> {
* \n* return runningFold(init(initial), operation)\n* \n*}\n* \n*\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* * to each element and current accumulator value that starts with [initial] value.\n* \n* * Note that `acc` value passed to [operation] function should not be mutated;\n* * otherwise it would affect the previous value in resulting list.\n* \n* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n* \n* * @sample samples.collections.Collections.Aggregates.scan

*\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* @kotlin.internal.InlineOnly\n* public inline fun <R> ULongArray.scan(initial: R, operation: (acc: R, ULong) -> R): List<R> {
* \n* return runningFold(init(initial), operation)\n* \n*}\n* \n*\n* Returns a list containing successive accumulation values generated by applying [operation] from left to right\n* * to each element and current accumulator value that starts with [initial] value.\n* \n* * Note that `acc` value passed to [operation] function should not be mutated;\n* * otherwise it would affect the previous value in resulting list.\n* \n* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.\n* \n* * @sample samples.collections.Collections.Aggregates.scan

Open Source Used In DNAC Platform Arica 9763
accumulator value and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.scan(initial: R, operation: (acc: R, UByte) -> R): List<R>
{
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
{
    return runningFold(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R>
{
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
{
    return runningFoldIndexed(initial, operation)
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
{
    return runningFoldIndexed(initial, operation)
}
```
```kotlin
/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumBy(selector: (ULong) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumBy(selector: (UByte) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumBy(selector: (UShort) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

function applied to each element in the array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UIntArray).sumByDouble(selector: (UInt) -> Double): Double {\n    var sum: Double = 0.0 for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(ULongArray).sumByDouble(selector: (ULong) -> Double): Double {\n    var sum: Double = 0.0 for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UByteArray).sumByDouble(selector: (UByte) -> Double): Double {\n    var sum: Double = 0.0 for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UShortArray).sumByDouble(selector: (UShort) -> Double): Double {\n    var sum: Double = 0.0 for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@Suppress("INAPPLICABLE_JVM_NAME")\@kotlin.jvm.JvmName("sumOfDouble")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UIntArray).sumOf(selector: (UInt) -> Double): Double {\n    var sum: Double = 0.toDouble for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@Suppress("INAPPLICABLE_JVM_NAME")\@kotlin.jvm.JvmName("sumOfDouble")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(ULongArray).sumOf(selector: (ULong) -> Double): Double {\n    var sum: Double = 0.toDouble for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@Suppress("INAPPLICABLE_JVM_NAME")\@kotlin.jvm.JvmName("sumOfDouble")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UByteArray).sumOf(selector: (UByte) -> Double): Double {\n    var sum: Double = 0.toDouble for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@Suppress("INAPPLICABLE_JVM_NAME")\@kotlin.jvm.JvmName("sumOfDouble")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UShortArray).sumOf(selector: (UShort) -> Double): Double {\n    var sum: Double = 0.toDouble for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\@Suppress("INAPPLICABLE_JVM_NAME")\@kotlin.jvm.JvmName("sumOfInt")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
(UIntArray).sumOf(selector: (UInt) -> Int): Int {\n    var sum: Int = 0.toInt for (element in this) {\n        sum += selector(element)  \n    return sum\n}  * Returns the sum of all values produced by [selector] function applied to each element in the array.

*\@SinceKotlin("1.4")\@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfInt")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfInt")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfInt")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Int): Int {
    var sum: Int = 0.toInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfLong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfLong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfLong")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> Long): Long {
    var sum: Long = 0.toLong()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in
 * the array.
 */
@SinceKotlin("1.4")\n@OptIn(kotlin.experimental.ExperimentalTypeInference::class)\n@OverloadResolution
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfUInt")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0.toUInt()
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
ByLambdaReturnType\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfUInt")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> UInt): UInt {\n    var sum = 0.toUInt()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfUInt")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> UInt): UInt {\n    var sum = 0.toUInt()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfUInt")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> UInt): UInt {\n    var sum = 0.toUInt()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sumOf(selector: (UInt) -> ULong): ULong {\n    var sum = 0.toULong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sumOf(selector: (ULong) -> ULong): ULong {\n    var sum = 0.toULong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sumOf(selector: (UByte) -> ULong): ULong {\n    var sum = 0.toULong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
ByLambdaReturnType@OptIn(kotlin.experimental.ExperimentalTypeInference::class)@OverloadResolution\n@Suppress("INAPPLICABLE_JVM_NAME")\n@kotlin.jvm.JvmName("sumOfULong")\n@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sumOf(selector: (UShort) -> ULong): ULong {\n    var sum = 0.toULong()\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n} * Returns the sum of all values produced by [selector] function applied to each element in the array.\n
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n} * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n* The returned list has length of the shortest collection.\n* @sample collections.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n} * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n* The returned list has length of the shortest collection.\n* @sample collections.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic infix fun <R> UByteArray.zip(other: UByteArray) {\n    return zip(other) { t1, t2 -> t1 to t2 }\n} * Returns a list of
pairs built from the elements of `this` array and the [other] array with the same index.\n
* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> UByteArray.zip(other: Array<out R>): List<Pair<UByte, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> UShortArray.zip(other: Array<out R>): List<Pair<UShort, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterableWithTransform\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterableWithTransform\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterableWithTransform\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public inline fun <R, V> UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterableWithTransform\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> UIntArray.zip(other: Iterable<R>): List<Pair<UInt, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> ULongArray.zip(other: Iterable<R>): List<Pair<ULong, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> UByteArray.zip(other: Iterable<R>): List<Pair<UByte, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable

*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun <R> UShortArray.zip(other: Iterable<R>): List<Pair<UShort, R>> \n{\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
*\n* The returned list has length of the shortest collection.\n* \n* @sample samples.collections.Operations.zipIterable
built from the elements of `this` collection and [other] array with the same index.

The returned list has length of the shortest collection.

@sample samples.collections.Iterables.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> UShortArray.zip(other: UShortArray): List<Pair<UShort, R>>
{ 
    return zip(other) { t1, t2 -> t1 to t2 } } // **

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> UByteArray.zip(other: UByteArray): List<Pair<UByte, R>>
{ 
    return zip(other) { t1, t2 -> t1 to t2 } } // **

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> ULongArray.zip(other: ULongArray): List<Pair<ULong, R>>
{ 
    return zip(other) { t1, t2 -> t1 to t2 } } // **

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> UIntArray.zip(other: UIntArray): List<Pair<UInt, R>>
{ 
    return zip(other) { t1, t2 -> t1 to t2 } } // **
List<Pair<UShort, UShort>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}\n
/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * @param samples\n * @param collections.\n * @param Iterables.\n * @param Operations.\n * @param zipWithTransform\n *\n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UIntArray.zip(other: UIntArray, transform: (a: UInt, b: UInt) -> V): List<V> {\n    val size = minOf(size,\n            other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * @param samples\n * @param collections.\n * @param Iterables.\n * @param Operations.\n * @param zipWithTransform\n *\n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> ULongArray.zip(other: ULongArray, transform: (a: ULong, b: ULong) -> V): List<V> {\n    val size = minOf(size,\n            other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * @param samples\n * @param collections.\n * @param Iterables.\n * @param Operations.\n * @param zipWithTransform\n *\n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UByteArray.zip(other: UByteArray, transform: (a: UByte, b: UByte) -> V): List<V> {\n    val size = minOf(size,\n            other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.\n * The returned list has length of the shortest array.\n * @param samples\n * @param collections.\n * @param Iterables.\n * @param Operations.\n * @param zipWithTransform\n *\n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <V> UShortArray.zip(other: UShortArray, transform: (a: UShort, b: UShort) -> V): List<V> {\n    val size = minOf(size,\n            other.size)\n    val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@kotlin.jvm.JvmName("sumOfUInt")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out UInt>.sum(): UInt {\n    var sum: UInt = 0u\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@kotlin.jvm.JvmName("sumOfULong")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out ULong>.sum(): ULong {\n    var sum: ULong = 0uL\n    for (element in this) {\n        sum += element\n    }\n    return sum\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@kotlin.jvm.JvmName("sumOfUByte")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out UByte>.sum(): UInt {\n    return sumBy { it.toUInt() }\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@kotlin.jvm.JvmName("sumOfUShort")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Array<out UShort>.sum(): UInt {\n    return storage.sum().toUInt()\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.sum(): UInt {\n    return storage.sum().toUInt()\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.sum(): ULong {\n    return storage.sum().toULong()\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.sum(): UInt {\n    return storage.sum().toUInt()\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.sum(): UInt {\n    return storage.sum().toUInt()\n}\n
/**
 * Returns the sum of all elements in the array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.sum(): UInt { 
    return sumBy { it.toUInt() } } 
*/

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UCollectionsKt")

package kotlin.collections;

/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * 
 * Generated by the GenerateStandardLib.kt
 * 
 * This file is AUTO-GENERATED by the GenerateStandardLib.kt
 * See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
 * 
 * Import kotlin.random
 * 
 * UShortArray
 *
 */

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an array of UByte containing all of the elements of this collection.
 */
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UCollectionsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns the sum of all elements in the collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UByte>.toUByteArray(): UByteArray {
    val result = UByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UInt>.toUIntArray(): UIntArray {
    val result = UIntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<ULong>.toULongArray(): ULongArray {
    val result = ULongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Collection<UShort>.toUShortArray(): UShortArray {
    val result = UShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

*@

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * 
 * This file is AUTO-GENERATED by the GenerateStandardLib.kt
 * See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
 * 
 * Import kotlin.random
 * 
 * UShortArray
 *
 */

import kotlin.jvm.JvmMultifileClass
import kotlin.jvm.JvmName("UComparisonsKt")

package kotlin.comparisons;

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt): UInt {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(a, maxOf(b, c))
}

*/

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * 
 * This file is AUTO-GENERATED by the GenerateStandardLib.kt
 * See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
 * 
 * Import kotlin.random
 * 
 * UShortArray
 *
 */

import kotlin.jvm.JvmMultifileClass
import kotlin.jvm.JvmName("UComparisonsKt")

package kotlin.comparisons;

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt): UInt {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

*/
greater of three values.

*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun maxOf(a: UByte, b: UByte, c: UByte): UByte {\n var max = a\n for (e in other) max = maxOf(max, e)\n return max\n} \n
*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun maxOf(a: UShort, b: UShort, c: UShort): UShort {\n var max = a\n for (e in other) max = maxOf(max, e)\n return max\n} \n
*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@public fun maxOf(a: UInt, vararg other: UInt): UInt {\n var max = a\n for (e in other) max = maxOf(max, e)\n return max\n} \n
*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@public fun maxOf(a: ULong, vararg other: ULong): ULong {\n var max = a\n for (e in other) max = maxOf(max, e)\n return max\n} \n
*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@public fun minOf(a: UInt, b: UInt): UInt {\n return if (a <= b) a else b\n} \n
*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@public fun minOf(a: ULong, b: ULong): ULong {\n return if (a <= b) a else b\n} \n
*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@public fun minOf(a: UByte, b: UByte): UByte {\n return if (a <= b) a else b\n} \n
*\@SinceKotlin("1.3\")\@ExperimentalUnsignedTypes\@public fun minOf(a: UShort, b: UShort): UShort {\n return if (a <= b) a else b\n} \n
*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@public fun minOf(a: UInt, vararg other: UInt): UInt {\n var min = a\n for (e in other) min = minOf(min, e)\n return min\n} \n
*\@SinceKotlin("1.4\")\@ExperimentalUnsignedTypes\@public fun minOf(a: ULong, vararg other: ULong): ULong {\n var min = a\n for (e in other) min = minOf(min, e)\n return min\n} \n
\nCopyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n* https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
progression from this value down to the specified [to] value with the step -1.\n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n *
 *\n *\n *
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun UByte.downTo(to: UByte):
UIntProgression {\n    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.\n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun UInt.downTo(to: UInt):
UIntProgression {\n    return UIntProgression.fromClosedRange(this, to, -1)\n}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.\n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun ULong.downTo(to: ULong):
ULongProgression {\n    return ULongProgression.fromClosedRange(this, to, -1L)\n}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.\n * The [to] value should be less than or equal to `this` value.\n * If the [to] value is greater than `this` value the returned progression is empty.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun UShort.downTo(to: UShort):
UIntProgression {\n    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public fun UIntProgression.reversed():
UIntProgression {\n    return UIntProgression.fromClosedRange(last, first, -step)\n}

/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public fun ULongProgression.reversed():
ULongProgression {\n    return ULongProgression.fromClosedRange(last, first, -step)\n}

/**
 * Returns a progression that goes over the same range with the given step.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun UIntProgression.step(step: Int):
UIntProgression {\n    checkStepIsPositive(step > 0, step)\n    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

/**
 * Returns a progression that goes over the same range with the given step.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes\n@public infix fun ULongProgression.step(step: Long):
ULongProgression {\n    checkStepIsPositive(step > 0, step)\n    return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

/**
 * Returns a range from this value up to but excluding the specified [to] value.\n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun UInt.until(to: UInt): UIntRange {\n    if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY\n    return this .. (to - 1u)\n}

/**
 * Returns a range from this value up to but excluding the specified [to] value.\n * If the [to] value is less than or equal to `this` value, then the returned range is empty.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public infix fun ULong.until(to: ULong): ULongRange {\n    if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY\n    return this .. (to - 1u)\n}

/**
 * Ensures that this value is not less than the specified 
 * [minimumValue].\n * @return \n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public fun UInt.coerceAtLeast(minimumValue: UInt):
UInt {\n    return if (this <= UInt.MIN_VALUE) minimumValue else this\n}

/**
 * Ensures that this value is not less than the specified 
 * [minimumValue].\n * @return \n *\n *\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@public fun ULong.coerceAtLeast(minimumValue: ULong):
ULong {\n    return if (this <= ULong.MIN_VALUE) minimumValue else this\n}
than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtLeastUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtLeast(minimumValue: ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

 /**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtLeastUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtLeast(minimumValue: UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

 /**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue]
or the [minimumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtLeastUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtLeast(minimumValue: UShort): UShort {
    return if (this < minimumValue) minimumValue else this
}

 /**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue]
or the [maximumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtMostUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtMost(maximumValue: UInt): UInt {
    return if (this > maximumValue) maximumValue else this
}

 /**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue]
or the [maximumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtMostUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtMost(maximumValue: ULong): ULong {
    return if (this > maximumValue) maximumValue else this
}

 /**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue]
or the [maximumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtMostUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

 /**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue]
or the [maximumValue] otherwise.

 samples.comparisons.ComparableOps.coerceAtMostUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtMost(maximumValue: UShort): UShort {
    return if (this > maximumValue) maximumValue else this
}

 /**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or minimumValue if this value is less than minimumValue,
or maximumValue if this value is greater than maximumValue.

 samples.comparisons.ComparableOps.coerceInUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    return if (this < minimumValue) minimumValue else if (this > maximumValue) maximumValue else this
}

 /**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue],
or [maximumValue] if this value is greater than [maximumValue].

 samples.comparisons.ComparableOps.coerceInUnsigned

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    return if (this < minimumValue) minimumValue else if (this > maximumValue) maximumValue else this
}
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

if (this < minimumValue) return minimumValue

if (this > maximumValue)

return maximumValue

return this

* Ensures that this value lies in the specified range

[minimumValue .. [maximumValue]].

* @return this value if it's in the range, or [maximumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {

if (minimumValue > maximumValue) throw

IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

if (this < minimumValue) return minimumValue

if (this > maximumValue)

return maximumValue

* Ensures that this value lies in the specified range [minimumValue .. [maximumValue]].

* @return this value if it's in the range, or 'range.start' if this value is less than 'range.start', or 'range.endInclusive' if this value is greater than 'range.endInclusive'.

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {

if (minimumValue > maximumValue) throw

IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")

if (this < minimumValue) return minimumValue

if (this > maximumValue)

return maximumValue

* Ensures that this value lies in the specified range [range].

* @return this value if it's in the [range], or 'range.start' if this value is less than 'range.start', or 'range.endInclusive' if this value is greater than 'range.endInclusive'.

* @sample samples.comparisons.ComparableOps.coerceInUnsigned

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
public fun UInt.coerceIn(range: ClosedRange<UInt>): UInt {

if (range is ClosedFloatingPointRange) {

return this.coerceIn<UInt>(range)

}

if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")

return

when {

this < range.start -> range.start

this > range.endInclusive -> range.endInclusive

else -> this

}

* /* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @file:kotlin.jvm.JvmMultifileClass
* @file:kotlin.jvm.JvmName("USequencesKt")
* @package kotlin.sequences

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.

 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UInt>.sum(): UInt {

var sum: UInt = 0u

for (element in this) {

sum += element

}

return sum

}

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.

 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<ULong>.sum(): ULong {

var sum: ULong = 0uL

for (element in this) {

sum += element

}

return sum

}

/**
 * Returns the sum of all elements in the sequence.
 *
 * The operation is _terminal_.

 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UByte>.sum(): UByte {

var sum: UByte = 0uB

for (element in this) {

sum += element

}

return sum

}
Sequence<UByte>.sum(): UInt { \n var sum = 0u\n for (element in this) \n return sum += element\n }\n
/**
 * Returns the sum of all elements in the sequence.\n * The operation is _terminal_.\n */\n@kotlin.jvm.JvmName("sumOfUShort")\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun Sequence<UShort>.sum(): UInt { \n var sum = 0u\n for (element in this) \n return sum += element\n }


/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */

package kotlin

public expect open class Error : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public

expect open class Exception : Throwable {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class RuntimeException : Exception {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IllegalArgumentException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IllegalStateException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public expect open class IndexOutOfBoundsException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class ConcurrentModificationException : RuntimeException {
    constructor()
    constructor(message: String?)
    @Deprecated("The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
    constructor(message: String?, cause: Throwable?)
    @Deprecated("The constructor is not supported on all platforms and will be removed from kotlin-stdlib-common soon.", level = DeprecationLevel.ERROR)
    constructor(cause: Throwable?)
}

public 

expect open class UnsupportedOperationException : RuntimeException {
    constructor()
    constructor(message: String?)
    constructor(message: String?, cause: Throwable?)
    constructor(cause: Throwable?)
}

public 

expect open class NumberFormatException : IllegalArgumentException {
    constructor()
}

public expect open class NullPointerException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public 

expect open class ClassCastException : RuntimeException {
    constructor()
    constructor(message: String?)
}

public expect open class 

AssertionError : Error {
    constructor()
    constructor(message: Any?)
}

/**
 * Thrown after invocation of a function or property that was expected to return `Nothing`,
 * but returned something instead.
 */
@SinceKotlin("1.4")
@PublishedApi
internal class KotlinNothingValueException : RuntimeException {
    constructor() : super()
    constructor(message: String?) : super(message)
    constructor(message: String?, cause: Throwable?) : super(message, cause)
    constructor(cause: Throwable?) : super(cause)
}

/**
 * Returns the detailed description of this throwable
 * with its stack trace.
 * The detailed description includes:
 * - the short description (see [Throwable.toString])
 * - the complete stack trace
 * - detailed descriptions of the exceptions that were [suppressed][suppressedExceptions] in order to deliver this exception
 * - the detailed description of each throwable in the [Throwable.cause] chain
 */
@SinceKotlin("1.4")
public expect fun
Open Source Used In DNAC Platform Arica 9779

```kotlin
Throwable.stackTraceToString(): String

/**
 * Prints the [detailed description][Throwable.stackTraceToString]
 * of this throwable to the standard output or standard error output.
 */
fun Throwable.printStackTrace(): Unit

/**
 * Gives a declaration (a function, a property or a class)
 * specific name in JavaScript.
 */
@Target(CLASS, FUNCTION, PROPERTY)
@OptionalExpectation
public expect annotation class JsName(val name: String)

/**
 * Marks experimental JS export annotations.
 */
@Experimental(level = Experimental.Level.WARNING)
@RequiresOptIn(level = RequiresOptIn.Level.WARNING)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.4")
public annotation class ExperimentalJsExport

package kotlin.js

import kotlin.annotation.AnnotationTarget.*

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that can
 * be iterated over.
 * @param T the type of element being iterated over. The iterator is covariant in its element type.
 */
public interface Iterable<out T> {
    operator fun iterator(): Iterator<T>
}

package kotlin.collections

import kotlin.internal.PlatformDependent

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that
 * can be iterated over and that supports removing elements during iteration.
 * @param T the type of element being iterated over. The iterator is covariant in its element type.
 */
public interface MutableIterable<out T> : Iterable<T> {
    override fun iterator():MutableIterator<T>
}
```

---

These annotations will be reported as warnings unless an explicit opt-in within * - if the [OptIn] annotation, e.g.
```
@OptIn(ExperimentalJsExport::class) println * or with the `-Xopt-in=kotlin.js.ExperimentalJsExport` compiler option
```
given.
```
This annotation is experimental, meaning that restrictions mentioned above are subject to change.
```

---

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

public interface MutableList<E> : List<E>, MutableCollection<E> {
    // Modification Operations
    /**
     * Adds all of the elements of the specified collection to
     * the result of this operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    /**
     * Adds the specified element to the collection.
     */
    override fun add(element: E): Boolean

    override fun remove(element: E): Boolean

    /**
     * Removes a single instance of the specified element from the
     * collection, if it is present.
     */
    override fun remove(element: E): Boolean

    /**
     * Removes all of the collection's elements that are also contained in the specified collection.
     * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.
     */
    public fun removeAll(elements: Collection<E>): Boolean

    /**
     * Removes all elements from this collection.
     */
    public fun clear(): Unit

    /**
     * Removes all elements from the collection that are contained in the specified collection.
     * @return `true` if any element was removed from the collection, `false` if the collection was not modified.
     */
    public fun retainAll(elements: Collection<E>): Boolean

    // Bulk Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
    override fun isEmpty(): Boolean
    override fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to
     * the end of this list.
     * @return `true` if the list was changed as the result of the operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    // Positional Access Operations
    override fun get(index: Int): E

    override fun listIterator(index: Int): ListIterator<E>

    // Search Operations
    public fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
    override fun isEmpty(): Boolean
    override fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to
     * the end of this list.
     * @return `true` if the list was changed as the result of the operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    // Positional Access Operations
    override fun get(index: Int): E

    // Search Operations
    public fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
    override fun isEmpty(): Boolean
    override fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to
     * the end of this list.
     * @return `true` if the list was changed as the result of the operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    // Positional Access Operations
    override fun get(index: Int): E

    // Search Operations
    public fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
    override fun isEmpty(): Boolean
    override fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to
     * the end of this list.
     * @return `true` if the list was changed as the result of the operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    // Positional Access Operations
    override fun get(index: Int): E

    // Search Operations
    public fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
    override fun isEmpty(): Boolean
    override fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to
     * the end of this list.
     * @return `true` if the list was changed as the result of the operation.
     */
    override fun addAll(elements: Collection<E>): Boolean

    // Positional Access Operations
    override fun get(index: Int): E

    // Search Operations
    public fun contains(element: @UnsafeVariance E): Boolean
    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Adds the specified element to the end of this list.
     */
    override fun add(element: E): Boolean

    // Query Operations
public val value: V
    }
}

/**
 * A modifiable collection that holds pairs of objects (keys and values) and
 * supports efficiently retrieving
 * the value corresponding to each
 * key. Map keys are unique; the map holds only one value for each key. Map keys are
 * unique; the map holds only one value for each key.
 * Methods in this interface support only read-only access to the set; read/write access is supported
 * through the [MutableSet] interface.
 *
 * @param E the type of elements contained in the set. The set is covariant in
 * its element type, as it's a generic unordered collection of elements that does not
 * support duplicate elements, and supports adding and removing elements. * @param E the type of elements contained in the set. The set is covariant in
 * its element type, as it's a generic unordered collection of elements that does not
 * support duplicate elements, and supports adding and removing elements.
 *
 * @param V the type of map values. The map is covariant in its value type,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
 * and supports efficient retrieval of key/value pairs in the map.
 * If the map is empty, `false` if the map is empty, `false` if the map is empty,
supports efficiently retrieving the value corresponding to each key. Map keys are unique; the map holds only one value for each key. The map is invariant in its key type. The map is invariant in its value type. The mutable map is invariant in its value type. Associates the specified value with the specified key in the map.

```java
public final fun put(key: K, value: V): V?
```
override fun hashCode(): Int =
    if (isEmpty()) -1 else (31 * (31 * first + last) +
    step)"/"**

    companion object {
        /**
         * Creates IntProgression within the specified bounds of a closed range.
         * The progress starts with the [rangeStart] value and goes toward the [rangeEnd] value not
         * excluding it, with the specified [step]. In order to go backwards the [step] must be negative.
         * [step] must be greater than 'Int.MIN_VALUE' and not equal to zero.
         */
    
    private val int init {\n init if (step == 0L) throw kotlin.IllegalArgumentException("\Step must be non-

    companion object {
        /**
         * Creates LongProgression within the specified bounds of a closed range.
         * The progress starts with the [rangeStart] value and goes toward the [rangeEnd] value not
         * excluding it, with the specified [step]. In order to go backwards the [step] must be negative.
         * [step] must be greater than 'Long.MIN_VALUE' and not equal to zero.
         */
    
    safe...
License. */

package kotlin.annotation

import kotlin.annotation.AnnotationTarget.*

/**
 * Contains the list of code elements which are the possible annotation targets.
 * */
public enum class AnnotationTarget {
    CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
    TYPE,
    EXPRESSION,
    FILE,
    TYPEALIAS
}

/**
 * Contains the list of possible annotation's retentions.
 * */
public enum class AnnotationRetention {
    SOURCE,
    BINARY,
    RUNTIME
}

/**
 * This meta-annotation indicates the kinds of code elements which are possible targets of an annotation.
 * @property allowedTargets list of allowed annotation targets
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Target(vararg val allowedTargets: AnnotationTarget)

/**
 * This meta-annotation determines whether an annotation is stored in binary output and visible for reflection (default retention)
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class Retention(val value: AnnotationRetention = AnnotationRetention.RUNTIME)

/**
 * This meta-annotation determines that an annotation is applicable twice or more on a single code element
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class Repeatable

/**
 * This meta-annotation determines that an annotation is a part of public API and therefore should be included in the generated documentation for the element to which the annotation is applied.
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class MustBeDocumented

/**
 * Specifies that the corresponding type parameter is not used for unsafe operations such as casts or 'is' checks
 * That means it's completely safe to use generic types as argument for such parameter.
 */
@Target(AnnotationTarget.TYPE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
internal annotation class PureReifiable

/**
 * Specifies that the corresponding built-in method exists depending on platform.
 * Current implementation for JVM looks whether method with same JVM descriptor exists in the module JDK.
 * For example MutableMap.remove(K, V) available only if corresponding method is defined in JDK (i.e. for major versions >= 8)
 */
@Target(AnnotationTarget.FUNCTION)
@Retention(AnnotationRetention.BINARY)
internal annotation class PlatformDependent

/*
 * Copyright 2010-2016 JetBrains s.r.o.
 * Use of this source code is governed by the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.internal

private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b:
Long {\n    val mod = a % b
    return if (mod >= 0) mod else mod + b
}\n\n// (a - b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int = if (a >= b) a - b else a + c - b

private fun differenceModulo(a: Long, b: Long, c: Long): Long = (a - b) mod c

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range \( \text{start} \) to \( \text{end} \) in case of a positive \([\text{step}]\), or from \( \text{end} \) to \( \text{start} \) in case of a negative \([\text{step}]\).
 * @param start the first element of the progression
 * @param end the end of the progression
 * @param step the increment of the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range \( \text{start} \) to \( \text{end} \) in case of a positive \([\text{step}]\), or from \( \text{end} \) to \( \text{start} \) in case of a negative \([\text{step}]\).
 * @param start the first element of the progression
 * @param end the end of the progression
 * @param step the increment of the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

@JsName("arrayIterator")
internal fun arrayIterator(array: dynamic, type: String?) = when (type) {
    null -> {
        val arr: Array<dynamic> = array
        object : Iterator<dynamic> {
            var index = 0
            override fun hasNext() = index < arr.size
            override fun next() = if (index < arr.size) arr[index++]
        }
    }
    "BooleanArray" -> booleanArrayIterator(array)
    "ByteArray" -> byteArrayIterator(array)
    "ShortArray" -> shortArrayIterator(array)
    "CharArray" -> charArrayIterator(array)
    "IntArray" -> intArrayIterator(array)
    "LongArray" -> longArrayIterator(array)
    "FloatArray" -> floatArrayIterator(array)
    "DoubleArray" -> doubleArrayIterator(array)
    else -> throw IllegalStateException("Unsupported type argument for arrayIterator:
    type = $type")
}

@JsName("booleanArrayIterator")
internal fun booleanArrayIterator(array: BooleanArray) = object : BooleanIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextBoolean() = if (index < array.size) array[index++]
}

@JsName("byteArrayIterator")
internal fun byteArrayIterator(array: ByteArray) = object : ByteIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextByte() = if (index < array.size) array[index++]
}

@JsName("shortArrayIterator")
internal fun shortArrayIterator(array: ShortArray) = object : ShortIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextShort() = if (index < array.size) array[index++]
}

@JsName("charArrayIterator")
internal fun charArrayIterator(array: CharArray) = object : CharIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextChar() = if (index < array.size) array[index++]
}

@JsName("intArrayIterator")
internal fun intArrayIterator(array: IntArray) = object : IntIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextInt() = if (index < array.size) array[index++]
}

@JsName("floatArrayIterator")
internal fun floatArrayIterator(array: FloatArray) = object : FloatIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextFloat() = if (index < array.size) array[index++]
}

@JsName("doubleArrayIterator")
internal fun doubleArrayIterator(array: DoubleArray) = object : DoubleIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextDouble() = if (index < array.size) array[index++]
}

// (a % b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

// (a % b) mod c
private fun differenceModulo(a: Long, b: Long, c: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

// (a % b) mod c
private fun
inline fun <T> success(value: T): Result<T> =
    Result(value)

/**
 * Returns an instance that encapsulates the given [Throwable] exception as failure.
 */
@Suppress("INAPPLICABLE_JVM_NAME")
@InlineOnly
@JvmName("failure")
public inline fun <T> failure(exception: Throwable): Result<T> =
    Result(createFailure(exception))

internal class Failure(
    @JvmField
    val exception: Throwable
) : Serializable {
    override fun equals(other: Any?): Boolean = other is Failure && exception == other.exception
    override fun hashCode(): Int = exception.hashCode()
    override fun toString(): String = "Failure($exception)"
}

/**
 * Creates an instance of internal marker [Result.Failure] class to
 * make sure that this class is not exposed in ABI.
 */
@PublishedApi
@SinceKotlin("1.3")
internal fun createFailure(exception: Throwable): Any =
    Result.Failure(exception)

/**
 * Throws exception if the result is failure. This internal function minimizes
 * inlined bytecode for [getOrThrow] and makes sure that in the future we can
 * add some exception-augmenting logic here (if needed).
 */
@PublishedApi
@SinceKotlin("1.3")
internal fun Result<*>.throwOnFailure() {
    if (value is Result.Failure) throw value.exception
}

/**
 * Calls the specified function [block] with 'this' as its receiver and returns its
 * encapsulated result if invocation was successful,
 * otherwise encapsulates the given [Throwable] exception.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T, R> T.runCatching(block: T.() -> R): Result<R> {
    return try {
        Result.success(block())
    } catch (e: Throwable) {
        Result.failure(e)
    }
}

/**
 * Calls the specified function [block] and returns its
 * encapsulated result if invocation was successful,
 * otherwise encapsulates the given [Throwable] exception.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R> runCatching(block: () -> R): Result<R> {
    return try {
        Result.success(block())
    } catch (e: Throwable) {
        Result.failure(e)
    }
}

// extensions ---

/**
 * Returns the encapsulated value if this instance represents [Success] or
 * throws the encapsulated [Throwable] exception if it is [Failure].
 * This function is a
 * shorthand for `getOrElse { throw it }` (see [getOrElse]).
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.getOrThrow(): T {
    throwOnFailure()
    return value as T
}

/**
 * Returns the
 * encapsulated value if this instance represents [Success] or the
 * result of [onFailure] function for
 * the encapsulated [Throwable] exception if it is [Failure].
 * Note, that this function
 * rethrows any [Throwable] exception thrown by [onSuccess] function.
 * This function is a shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T, R> Result<T>.orElse(onFailure: (exception: Throwable) -> R): R {
    return when (val exception = exceptionOrNull()) {
        null -> onSuccess
        else -> onFailure(exception)
    }
}

/**
 * Returns the
 * encapsulated value if this instance represents [Success] or the
 * default value if it is [Failure].
 * Note, that this function
 * rethrows any [Throwable] exception thrown by [onSuccess] or by [onFailure] function.
 * This function is a
 * shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T, R> Result<T>.getOrElse(defaultValue: R): R {
    if (isFailure) return defaultValue
    return value as T
}

/**
 * Returns the
 * result of [onSuccess] for the
 * encapsulated value if this instance represents [Success] or [onFailure] function for
 * the encapsulated [Throwable] exception if it is [Failure].
 * Note, that this function
 * rethrows any [Throwable] exception thrown by [onSuccess] or by [onFailure] function.
 * This function is a
 * shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T, R> Result<T>.fold(onSuccess: (value: T) -> R, onFailure: (exception: Throwable) -> R): R {
    return try {
        onSuccess
    } catch (e: Throwable) {
        onFailure(e)
    }
}

// transformation

/**
 * Returns the encapsulated result of the given
 * function applied to the encapsulated value
 * if this instance represents [Success] or
 * the original encapsulated [Throwable] exception if it is [Failure].
 * Note, that this function
 * rethrows any [Throwable] exception thrown by [transform] function.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> transform(value: T): Result<T> =
    Result(transform(value))

Open Source Used In DNAC Platform Arica  9780
Result<T>.map(transform: (value: T) -> R): Result<R> {
    contract {
        callsInPlace(transform, InvocationKind.AT_MOST_ONCE)
    } return when {
        isSuccess -> Result.success(transform(value as T))
        else -> Result(value)
    }.*
}

/**
 * Returns the encapsulated result of the given [transform] function applied to the encapsulated value if this instance represents [success]; otherwise returns the original value.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T, R> Continuation<T>.resume(value: T): Unit =
    resumeWith(Result.success(value))

/**
 * Resumes the execution of the corresponding coroutine passing a successful or failed [result] as the value of the last suspension point.
 */
@SinceKotlin("1.3")
public interface Continuation

/**
 * Interface representing a continuation after a suspension point that returns a value of type `T`.
 */
@SinceKotlin("1.3")
public interface Continuation<T extends Any> {
    /**
     * The context of the coroutine that corresponds to this continuation.
     */
    public val context: CoroutineContext

    /**
     * Resumes the execution of the corresponding coroutine passing a successful or failed [result] as the value of the last suspension point.
     */
    public fun resumeWith(result: Result<T>): Result<T>*
}

/**
 * Resumes the

---

"Open Source Used In DNAC Platform Arica" 9791
execution of the corresponding coroutine so that the [exception] is re-thrown right after the
last suspension point.

```kotlin
Continuation<T>.resumeWithException(exception: Throwable): Unit =
resumeWith(Result.failure(exception))

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
```

Open Source Used In DNAC Platform Arica 9792
Starts an unintercepted coroutine without a receiver and with result type \([T]\) and executes it until its first suspension.

* Returns the result of the coroutine or throws its exception if it does not suspend or \([\text{COROUTINE\_SUSPENDED}]\) if it suspends.

* In the latter case, the \([\text{completion}]\) continuation is invoked when the coroutine completes with a result or an exception.

* The coroutine is started directly in the invoker's thread without going through the \([\text{ContinuationInterceptor}]\) that might\(n\) * be present in the completion's \([\text{CoroutineContext}]\).

It is the invoker's responsibility to ensure that a proper invocation context is established.

* This function is designed to be used from inside of \([\text{suspendCoroutineUninterceptedOrReturn}]\) to resume the execution of the suspended\(n\) * coroutine using a reference to the suspending function.

\[
\text{this.asDynamic()}(completion, false)\text{un}/**\text{un} * \text{Starts an unintercepted coroutine with receiver type \([R]\) and result type \([T]\) and executes it until its first suspension.}\text{un} * \text{Returns the result of the coroutine or throws its exception if it does not suspend or } [\text{COROUTINE\_SUSPENDED}] \text{ if it suspends.}\text{un} * \text{In the latter case, the } [\text{completion}] \text{ continuation is invoked when the coroutine completes with a result or an exception.}\text{un} * \text{The coroutine is started directly in the invoker's thread without going through the } [\text{ContinuationInterceptor}] \text{ that might\(n\) * be present in the completion's } [\text{CoroutineContext}] \text{. It is the invoker's responsibility to ensure that a proper invocation context is established.}\text{un} * \text{This function is designed to be used from inside of } [\text{suspendCoroutineUninterceptedOrReturn}] \text{ to resume the execution of the suspended\(n\) * coroutine using a reference to the suspending function.}\text{un} * \text{This function returns unintercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on the returned } [\text{continuation}] \text{ instance.}\text{un} * \text{The } [\text{completion}] \text{ continuation is invoked when coroutine completes with result or exception.}\text{un} * \text{This function returns unintercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on intercepted continuation guarantees that execution of}n * \text{both the coroutine and } [\text{completion}] \text{ happens in the invocation context established by}n * \text{[ContinuationInterceptor] that might be present in the completion's } [\text{CoroutineContext}].}\text{un} * \text{Note that } [\text{completion}] \text{ of this function may get invoked in an arbitrary context.}\text{un} * \text{[Continuation.intercepted] can be used to acquire the intercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on intercepted continuation guarantees that execution of}n * \text{both the coroutine and } [\text{completion}] \text{ happens in the invocation context established by}n * \text{[ContinuationInterceptor] that might be present in the completion's } [\text{CoroutineContext}].}\text{un} * \text{[Continuation.intercepted] can be used to acquire the intercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on intercepted continuation guarantees that execution of}n * \text{both the coroutine and } [\text{completion}] \text{ happens in the invocation context established by}n * \text{[ContinuationInterceptor] that might be present in the completion's } [\text{CoroutineContext}].}\text{un} * \text{It is the invoker's responsibility to ensure that a proper invocation context is established.}\text{un} * \text{Note that } [\text{completion}] \text{ of this function may get invoked in an arbitrary context.}\text{un} * \text{[Continuation.intercepted] can be used to acquire the intercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on intercepted continuation guarantees that execution of}n * \text{both the coroutine and } [\text{completion}] \text{ happens in the invocation context established by}n * \text{[ContinuationInterceptor] that might be present in the completion's } [\text{CoroutineContext}].}\text{un} * \text{It is the invoker's responsibility to ensure that a proper invocation context is established.}\text{un} * \text{Note that } [\text{completion}] \text{ of this function may get invoked in an arbitrary context.}\text{un} * \text{[Continuation.intercepted] can be used to acquire the intercepted continuation.}\text{un} * \text{Invocation of } [\text{resume(Unit)}] \text{ on intercepted continuation guarantees that execution of}n * \text{both the coroutine and } [\text{completion}] \text{ happens in the invocation context established by}n * \text{[ContinuationInterceptor] that might be present in the completion's } [\text{CoroutineContext}].}
coroutine and [completion] happens in the invocation context established by
this. [ContinuationInterceptor]. When suspended is true the continuation is created, but not executed
if this. [ContinuationInterceptor].re

val this = ContinuationInterceptor

private inline fun <T> createCoroutineFromSuspendFunction(
    completion: Continuation<T>,
    crossinline block: () -> Any?
): Continuation<Unit> {
    @Suppress("UNCHECKED_CAST")
    return object
        : CoroutineImpl(completion as Continuation<Any?>) {
            override fun doResume(): Any? {
                exception?.let { throw it }
                return block()
            }
        }
}

/**
 * Intercepts this continuation with [ContinuationInterceptor].
 *
 * This function shall be used on the immediate result of
 * [createCoroutineUnintercepted] or [suspendCoroutineUninterceptedOrReturn].
 * [Completion] functions invoke [ContinuationInterceptor.interceptContinuation], caches and returns the result.
 * If this function is invoked on other [Completion] instances it returns this continuation unchanged.
 * [Completion] hooks shall check for [ContinuationInterceptor] in the continuation's context and invoke
 * [Completion.intercepted] if this is the case.
 */
@SinceKotlin("1.3")
public actual fun <T> Continuation<T>.intercepted(): Continuation<T> =
    (this as? CoroutineImpl)?.intercepted() ?:
    this

private fun isArrayish(o: dynamic) =
    js("Kotlin").isArrayish(o)
Throwable?) : this(undef, cause)

public actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(undef, cause)
    this(undef, cause)
}

public actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message) {
    this(null)
}

public actual open class NullPointerException actual constructor(message: String?) : RuntimeException(message) {
    this(null)
}

public actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message) {
    this(null)
}

public actual open class AssertionError
@SinceKotlin("1.4")
constructor(message: String?, cause: Throwable?) : Error(message, cause) {
    this(null)
    constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(undef, cause)
}

public actual open class NoSuchElementException actual constructor(message: String?) : RuntimeException(message) {
    this(null)
}

@SinceKotlin("1.3")
public actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message) {
    this(null)
}

public actual open class NoWhenBranchMatchedException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(undef, cause)
}

public actual open class UninitializedPropertyAccessException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(undef, cause)
}

@file:Suppress("UNUSED_PARAMETER")

package kotlin.js

@kotlin.internal.InlineOnly
internal inline fun jsDeleteProperty(obj: Any, property: Any) {
    js("delete obj[property]")
}

@kotlin.internal.InlineOnly
internal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =
    js("lhs | rhs").unsafeCast<Int>()
to the IEEE 754 floating-point "single format" bit layout, * note that in Kotlin/JS [Float] range is wider than "single format" bit layout can represent, * so some [Float] values may overflow, underflow or lose their accuracy after conversion to bits and back.

`Float.toRawBits()` returns the [Float] value corresponding to a given bit representation.

```kotlin
@SinceKotlin("1.2")
@library("floatToRawBits")
public actual fun Float.toRawBits(): Int =
  definedExternally
```

```kotlin
/**
 * Returns the [Float] value corresponding to a given bit representation.
 */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun Float.Companion.fromBits(bits: Int): Float = js("Kotlin").floatFromBits(bits).unsafeCast<Float>()
```

```kotlin
internal inline fun Long(low: Int, high: Int) = js("Kotlin").Long.fromBits(low, high).unsafeCast<Long>()
internal inline val Long.low: Int get() =
  this.asDynamic().getLowBits().unsafeCast<Int>()
internal inline val Long.high: Int get() =
  this.asDynamic().getHighBits().unsafeCast<Int>()
```

```kotlin
@end_omit
```

```kotlin
@PublishedApi
internal fun <T : Annotation> KClass<*>.findAssociatedObject(annotationClass: KClass<T>): Any? {
    // This API is not supported in js-v1. Return `null` to be source-compatible with js-ir.
    return null
}
```

```kotlin
public actual fun Long.toString(radix: Int): String = asDynamic().toString(checkRadix(radix))
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

```kotlin
import kotlin.reflect.KClass

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import java.util.Arrays
import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Reverses elements in the list in-place.
 */
public actual fun <T> MutableList<T>.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import java.util.Arrays
import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.elementAt(index: Int): UInt {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}
```

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import java.util.Arrays
import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun ULongArray.elementAt(index: Int): ULong {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}
```
/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.elementAt(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.elementAt(index: Int): UShort {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.asList(): List<UInt> {
    return object : AbstractList<UInt>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UInt): Boolean = this@asList.contains(element)
        override fun get(index: Int): UInt {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:Suppress("DEPRECATION_ERROR")
package jquery.ui

import jquery.JQuery
import kotlin.js.Json

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.buttonset(): JQuery =
    asDynamic().buttonset()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.", level = DeprecationLevel.ERROR)
public inline fun JQuery.dialog(): JQuery =
file you tell the compiler to import a JavaScript object → that contain all the declarations from the file.

Example:

```kotlin
*S* *un* * `kotlin* * @JsModule("jquery")* * external abstract class JQuery(* * @un* * // some declarations here un * )un * @JsModule("jquery")* * external fun JQuery(element: Element): JQueryun * * @property import name of a module to import declaration from.un * It is not interpreted by the Kotlin compiler, it's passed as is directly to the target module system.un * @see JsNonModuleName
```

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
}
```

@JsModule annotation you can make a declaration unavailable to plain module system.

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
}
```

@JsModule annotation you can make a declaration unavailable to plain module system.

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
}
```

@JsModule annotation you can make a declaration unavailable to plain module system.

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
}
```

@JsModule annotation you can make a declaration unavailable to plain module system.

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
}
```

@JsModule annotation you can make a declaration unavailable to plain module system.
/**
 * Creates an empty [ArrayList].
 */
    public actual constructor() : this(emptyArray()) {}

    /**
     * Creates an empty [ArrayList].
     * @param initialCapacity initial capacity (ignored)
     */
    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual constructor(initialCapacity: Int = 0) : this(emptyArray()) {}

    /**
     * Creates an [ArrayList] filled from the [elements] collection.
     */
    public actual constructor(elements: Collection<E>) : this(elements.toTypedArray<Any?>()) {}

    @PublishedApi
    internal fun build(): List<E> {
        checkIsMutable()
        isReadOnly = true
        return this
    }

    /**
     * Does nothing in this ArrayList implementation.
     * @param minCapacity minimum capacity (not used)
     */
    public actual fun trimToSize() {}

    /**
     * Does nothing in this ArrayList implementation.
     * @param minCapacity minimum capacity (not used)
     */
    public actual fun ensureCapacity(minCapacity: Int) {}

    actual override val size: Int get() = array.size

    @Suppress("UNCHECKED_CAST")
    actual override fun get(index: Int): E = array[rangeCheck(index)] as E

    actual override fun set(index: Int, element: E): E {
        checkIsMutable()
        rangeCheck(index)
        @Suppress("UNCHECKED_CAST")
        return array[index].apply { array[index] = element } as E
    }

    actual override fun add(element: E): Boolean {
        checkIsMutable()
        array.asDynamic().push(element)
        modCount++
        return true
    }

    actual override fun add(index: Int, element: E): Unit {
        checkIsMutable()
        rangeCheck(index)
        @Suppress("UNCHECKED_CAST")
        array.asDynamic().splice(insertionRangeCheck(index), 0, element)
        modCount++
    }

    actual override fun addAll(elements: Collection<E>): Boolean {
        checkIsMutable()
        if (elements.isEmpty()) return false
        array += elements.toTypedArray<Any?>()
        modCount++
        return true
    }

    actual override fun addAll(index: Int, elements: Collection<E>): Boolean {
        checkIsMutable()
        insertionRangeCheck(index)
        when (index) {
            size -> addAll(elements)
            else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
        }
        modCount++
        return true
    }

    actual override fun removeAt(index: Int): E {
        checkIsMutable()
        rangeCheck(index)
        modCount++
        return if (index == lastIndex)
            array.asDynamic().pop()
        else
            array.asDynamic().splice(index, 1)[0]
    }

    actual override fun remove(element: E): Boolean {
        checkIsMutable()
        for (index in array.indices) {
            if (array[index] == element) {
                array.asDynamic().splice(index, 1)
                modCount++
                return true
            }
        }
        return false
    }

    override fun removeRange(fromIndex: Int, toIndex: Int) {
        checkIsMutable()
        modCount++
        array.asDynamic().splice(fromIndex, toIndex - fromIndex)
    }

    actual override fun clear() {
        checkIsMutable()
        array = emptyArray()
        modCount++
    }

    actual override fun indexOf(element: E): Int = array.indexOf(element)

    actual override fun lastIndexOf(element: E): Int = array.lastIndexOf(element)

    override fun toString() = arrayToString(array)
    override fun toArray(): Array<Any?> = js("["]").slice.call(array)

    internal override fun checkIsMutable() {
        if (isReadOnly)
            throw UnsupportedOperationException()
    }

    private fun rangeCheck(index: Int) = index.apply {
        AbstractList.checkElementIndex(index, size)
    }

    private fun insertionRangeCheck(index: Int) = index.apply {
        AbstractList.checkPositionIndex(index, size)
    }
}
array.asDynamic().sort(comparison))

else {
    mergeSort(array.unsafeCast<Array<T>>(), 0,
    array.lastIndex, naturalOrder())
}

private fun String.unsafeCast<Array<T>>()
    return transformArray<in T>(array, convert)

private fun <T> transformArray<in T>(array: Array<T>, convert: (T) -> T): Array<T>
    return with(array) { UnsafeWrap(this).unsafeCast<Array<T>>() }

private fun <T> mergeSort(array: Array<T>, start: Int,
endInclusive: Int, comparator: Comparator<in T>) {
    val buffer = arrayOfNulls<Any?>(array.size).unsafeCast<Array<T>>()
    val result = mergeSort(array, buffer, start,
endInclusive, comparator)
    if (result !== array) {
        for (i in start..endInclusive) array[i] = result[i]
    }
}

// Both start and end are inclusive indices.
private fun <T> mergeSort(array: Array<T>, buffer: Array<T>,
start: Int, end: Int, comparator: Comparator<in T>): Array<T> {
    if (start == end) {
        return array
    }

    val median = (start + end) / 2
    val left = mergeSort(array, buffer, start, median, comparator)
    val right = mergeSort(array, buffer, median + 1, end, comparator)

    val target = if (left === buffer) array else buffer
// Merge.
    var leftIndex = start
    var rightIndex = median + 1
    for (i in start..end) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                val leftValue = left[leftIndex]
                val rightValue = right[rightIndex]

                if (comparator.compare(leftValue, rightValue) <= 0) {
                    target[i] = leftValue
                    leftIndex++
                } else {
                    target[i] = rightValue
                    rightIndex++
                }
            }
            leftIndex <= median -> {
                target[i] = left[leftIndex]
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = right[rightIndex]
                rightIndex++
                Unit // TODO: Fix KT-31506
            }
        }
    }

    return target
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

@OptIn(ExperimentalUnsignedTypes::class)
@SinceKotlin("1.3")
@kotlin.js.JsName("contentDeepHashCodeImpl")
internal fun <T> Array<out T>?.contentDeepHashCodeImpl(): Int {
    if (this == null) return 0
    var result = 1
    for (element in this) {
        val elementHash = when {
            element == null -> 0
            isArrayish(element) -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()
            element is UByteArray   -> element.contentHashCode()
            element is UShortArray  -> element.contentHashCode()
            element is UIntArray    -> element.contentHashCode()
            element is ULongArray   -> element.contentHashCode()
            else                    -> element.hashCode()
        }

        result = 31 * result + elementHash
    }
    return result
}
AbstractMutableSet<MutableEntry<K, V>> {

    override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on entries")

    override fun clear() {
        this@HashMap.clear()
    }

    override operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)

    override operator fun iterator(): MutableIterator<MutableEntry<K, V>> = internalMap.iterator()

    override fun remove(element: MutableEntry<K, V>): Boolean {
        if (contains(element)) {
            this@HashMap.remove(element.key)
            return true
        }
        return false
    }

    override val size: Int get() = this@HashMap.size
}

/**
 * Internal implementation of the map: either string-based or hashcode-based.
 */
private val internalMap: InternalMap<K, V> private val equality: EqualityComparator

internal constructor(internalMap: InternalMap<K, V>) : super() {
    this.internalMap = internalMap
    this.equality = internalMap.equality
}

actual constructor() : this(InternalHashCodeMap(EqualityComparator.HashCode))

actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) {
    require(initialCapacity >= 0) { "Negative initial capacity: $initialCapacity" }
    require(loadFactor >= 0) { "Non-positive load factor: $loadFactor" }
}

actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

/**
 * Constructs an instance of [HashSet] with the contents of the specified [original] map.
 */
actual constructor(original: Map<out K, V>) : this() {
    this.putAll(original)
}

actual override fun clear() {
    internalMap.clear()
    //structureChanged(this)
}

actual override fun containsKey(key: K): Boolean = internalMap.containsKey(key)

actual override fun containsValue(value: V): Boolean = internalMap.any { it.value == value }

private var _entries: MutableSet<MutableMap.MutableEntry<K, V>>? = null
actual override val entries: MutableSet<MutableMap.MutableEntry<K, V>>
    get() {
        if (_entries == null) {
            _entries = createEntrySet()
        }
        return _entries!!
    }

protected open fun createEntrySet(): MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()

actual override fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
    return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply {
        putAll(pairs)
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/*
 * Based on GWT HashSet
 */

package kotlin.collections

/**
 * The implementation of the [MutableSet] interface, backed by a [HashMap] instance.
 */
// Classes that extend HashSet and implement `build()` (freezing) operation have to make sure mutating methods check `checkIsMutable`
private actual open class HashSet<E> : AbstractMutableSet<E>, MutableSet<E> {

    internal val map: HashMap<E, Any>

    /**
     * Constructs a new empty [HashSet].
     */
    actual constructor() {
        map = HashMap<E, Any>()
    }

    /**
     * Constructs a new [HashSet] filled with the elements of the specified collection.
     */
    actual constructor(elements: Collection<E>) {
        map = HashMap<E, Any>(elements.size)
        addAll(elements)
    }

    /*
     * Constructs a new [HashMap] with [String] keys, which stores the keys as properties of an * JS object without hashing them.
     */
    actual public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
        return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs) }
    }

    actual override fun clear() {
        //structureChanged(this)
    }

    actual override fun contains(element: Any): Boolean = containsKey(element)

    actual override fun containsAll(elements: Collection<Any>): Boolean = elements.all { contains(it) }

    actual override val size: Int get() = map.size
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/*
 * Based on GWT HashSet
 */

package kotlin.collections
underlying map. This is used by

    * LinkedHashSet.

    * @param map underlying map to use.

    * @param key the key to add.

    * @param value the value to add.

    * @param element the element to add.

    * @param this the this reference.

    * @param add adds the element to the map.

    * @param map the map to use.

    * @param key the key to clear.

    * @param value the value to clear.

    * @param element the element to clear.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.

    * @param key the key to return.

    * @param value the value to return.

    * @param element the element to return.

    * @param this the this reference.

    * @param map the map to use.
Array<MutableEntry<K, V>>.findEntryInChain(key: K): MutableEntry<K, V>? = 
override fun iterator(): MutableIterator<MutableEntry<K, V>> {
    val chainOrEntry = backingMap[hashCode]
    return if (chainOrEntry === undefined) null else 
        Array<MutableEntry<K, V>>.findEntryInChain(key, chainOrEntry)
}

private fun getChainOrEntryOrNull(hashCode: Int): dynamic {
    val chainOrEntry = backingMap[hashCode]
    return if (chainOrEntry === undefined) null else chainOrEntry
}

internal class InternalStringMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {

    private var backingMap: dynamic = createJsMap()
    override var size: Int = 0

    override operator fun contains(key: K): Boolean {
        return backingMap[key] !== undefined
    }

    override operator fun get(key: K): V? {
        return backingMap[key].unsafeCast<V>()
    }

    override fun put(key: K, value: V): V? {
        val oldValue = backingMap[key]
        backingMap[key] = value
        if (oldValue !== undefined) {
            return oldValue.unsafeCast<V>()
        } else {
            return null
        }
    }

    override fun remove(key: K): V? {
        val value = backingMap[key]
        if (value !== undefined) {
            jsDeleteProperty(backingMap, key)
            return value
        } else {
            return null
        }
    }
}

```kotlin
override fun clear() {
    backingMap = createJsMap()
    size = 0
}

override fun iterator():MutableIterator<MutableEntry<K, V>> =
    object:
        @Suppress("UNCHECKED_CAST")
        @Suppress("UNCHECKED_CAST")
        InternalStringMap.remove(checkNotNull(lastKey) as K)@suppress

private inner class ChainEntry<K, V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V>(key, value) {
        internal var next: ChainEntry<K, V>? = null
        internal var prev: ChainEntry<K, V>? = null

        override fun setValue(newValue: V): V {
            this@LinkedHashMap.checkIsMutable()
            return super.setValue(newValue)
        }
    }

private inner class EntrySet : AbstractMutableSet<MutableEntry<K, V>>() {

    private inner class EntryIterator : MutableIterator<MutableEntry<K, V>> {
        // The last entry that was returned from this iterator.
        private var last: ChainEntry<K, V>? = null

        // The next entry to return from this iterator.
        private var next: ChainEntry<K, V>? = null

        init {
            next = head
        }

        override fun hasNext(): Boolean {
            return next !== null
        }

        override fun next(): MutableEntry<K, V> {
            if (!hasNext()) throw NoSuchElementException()

            val current = next!!
            last = current
            next = current.next.takeIf { it !== head }
            return current
        }

        override fun remove() {
            check(last != null)
            this@EntrySet.checkIsMutable()
            last!!.remove()
            map.remove(last!!.key)@suppress
            if (hasNext()) throw NoSuchElementException()
        }
    }

    override fun add(element:MutableEntry<K, V>):Boolean = throw
    UnsupportedOperationException("Add is not supported on entries")

    override fun clear() {
        this@LinkedHashMap.clear()
        override operator fun contains(element:MutableEntry<K, V>):Boolean = containsEntry(element)
        override operator fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(entries)
        override operator fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsKey(key:K):Boolean = containsKey(key)
        override fun containsValue(value:V):Boolean = containsValue(value)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = containsAny(elements)
        override fun containsAll(elements:Collection<MutableEntry<K, V>>):Boolean = containsAll(elements)
        override fun containsAny(elements:Collection<MutableEntry<K, V>>):Boolean = contain
package kotlin.collections

/**
 * The implementation of the [MutableSet] interface, backed by a [LinkedHashMap] instance.
 */
public actual open class LinkedHashSet<E> : HashSet<E>, MutableSet<E> {

    internal constructor(map: LinkedHashMap<E, Any>) : super(map)

    /**
     * Constructs a new empty [LinkedHashSet].
     */
    actual constructor() : super(LinkedHashMap<E, Any>())

    /**
     * Constructs a new [LinkedHashSet] filled with the elements of the specified collection.
     */
    actual constructor(elements: Collection<E>) : super(LinkedHashMap<E, Any>()) {
        addAll(elements)
    }

    /**
     * Constructs an [LinkedHashSet] with the initial capacity and load factor.
     */
    actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(LinkedHashMap<E, Any>(initialCapacity, loadFactor))

    actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

    @PublishedApi
    internal fun build(): Set<E> {
        (map as LinkedHashMap<E, Any>).build()
        return this
    }

    internal override fun checkIsMutable(): Unit = map.checkIsMutable()

//    public override fun clone(): Any {
//        return LinkedHashSet(this)
//    }

}

/**
 * Creates a new instance of the specialized implementation of [LinkedHashSet] with the specified [String] elements,
 * which elements the keys as properties of JS object without hashing them.
 */
public fun linkedStringSetOf(vararg elements: String): LinkedHashSet<String> {

    return LinkedHashSet(linkedStringMapOf<Any>()).apply { addAll(elements) }
}

private fun String(value: Any?): String = js("String")(value)

/** Prints the line separator to the standard output stream. */
public actual fun println() {
    output.println()
}

/** Prints the given [message] and the line separator to the standard output stream. */
public fun println(message: Any?) {
    console.log(message)
    println()
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("NodeJsOutput")
internal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {
    override fun print(message: Any?) {
        // TODO: Using local variable because of bug in block decomposition lowering in IR backend
        val messageString = String(message)
        outputStream.write(messageString)
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("OutputToConsoleLog")
internal class OutputToConsoleLog : BaseOutput() {
    override fun println() {
        console.log("")
    }

    override fun println(message: Any?) {
        console.log(message)
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutput")
internal open class BufferedOutput : BaseOutput() {
    var buffer = ""

    override fun print(message: Any?) {
        buffer += String(message)
    }

    override fun flush() {
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {
    override fun print(message: Any?) {
        console.log(String(message))
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("output")
internal var output = run {
    val isNode: Boolean = js("typeof process !== 'undefined' && process.versions && !!process.versions.node")
    if (isNode) NodeJsOutput(js("process.stdout")) else BufferedOutputToConsoleLog()
}

/** Prints the line separator to the standard output stream. */
@JsName("println")
internal fun println() {
    output.println()
}

/** Prints the given [message] and the line separator to the standard output stream. */
@JsName("println")
internal fun println(message: String) {
    output.println(message)
    println()
actual fun println(message: Any?) {
    output.println(message)
}

/** Prints the given [message] to the standard output stream. */
public actual fun print(message: Any?) {
    output.print(message)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines

import kotlin.coroutines.intrinsics.CoroutineSingletons.*
import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED

@PublishedApi
@SinceKotlin("1.3")
internal actual class SafeContinuation<in T>
internal actual constructor(
    private val delegate: Continuation<T>,
    initialResult: Any?
) : Continuation<T> {
    @PublishedApi
    internal actual constructor(delegate: Continuation<T>) : this(delegate, UNDECIDED)

    public actual override val context: CoroutineContext
        get() = delegate.context

    private var result: Any? = initialResult

    public actual override fun resumeWith(result: Result<T>) {
        val cur = this.result
        when {
            cur === UNDECIDED -> {
                this.result = result.value
            }
            cur === COROUTINE_SUSPENDED -> {
                this.result = RESUMED
                delegate.resumeWith(result)
            }
            else -> throw IllegalStateException("Already resumed")
        }
    }

    @PublishedApi
    internal actual fun getOrThrow(): Any? {
        if (result === UNDECIDED) {
            result = COROUTINE_SUSPENDED
        }
        val result = this.result
        return when (result) {
            RESUMED -> COROUTINE_SUSPENDED
            upstream result is Result.Failure -> throw result.exception
            else -> return (result)
        }
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines.cancellation

@ExperimentalStdlibApi
@SinceKotlin("1.4")
public actual open class CancellationException : IllegalStateException {
    actual constructor() : super()
    actual constructor(message: String?) : super(message)
    constructor(message: String?, cause: Throwable?) : super(message, cause)
    constructor(cause: Throwable?) : super(cause)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines.js.internal

import kotlin.coroutines.Continuation
import kotlin.coroutines.EmptyCoroutineContext

@PublishedApi
@SinceKotlin("1.3")
internal val EmptyContinuation = Continuation<Any?>(EmptyCoroutineContext) { result ->
    result.getOrThrow()
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.js

/**
 * Exposes the [Date API](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public external class Date() {
    public constructor(milliseconds: Number)
    public constructor(dateString: String)
    public constructor(year: Int, month: Int)
    public constructor(year: Int, month: Int, day: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number)
    public fun getDate(): Int
    public fun getDay(): Int
    public fun getFullYear(): Int
    public fun getHours(): Int
    public fun getMilliseconds(): Int
    public fun getMinutes(): Int
    public fun getMonth(): Int
    public fun getSeconds(): Int
    public fun getTime(): Double
    public fun getTimezoneOffset(): Int
    public fun getUTCDate(): Int
    public fun getUTCDay(): Int
    public fun getUTCFullYear(): Int
    public fun getUTCHours(): Int
    public fun getUTCMilliseconds(): Int
    public fun getUTCMinutes(): Int
    public fun getUTCMonth(): Int
    public fun getUTCSeconds(): Int
    public fun toDateString(): String
    public fun toISOString(): String
    public fun toJSON(): Json
    public fun toLocaleDateString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toLocaleDateString(): String
    public fun toLocaleDateString(): String
    public fun toLocaleDateString(): String
    public fun toLocaleDateString(): String
}
LocaleOptions = definedExternally): String

public fun toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String

public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String

public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String

public fun toLocaleTimeString(locales: String, options: LocaleOptions = definedExternally): String

public fun toTimeString(): String

public fun toUTCString(): String

public companion object {

    public fun now(): Double

    public fun parse(dateString: String): Double

    public fun UTC(year: Int, month: Int): Double

    public fun UTC(year: Int, month: Int, day: Int): Double

    public fun UTC(year: Int, month: Int, day: Int, hour: Int): Double

    public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int): Double


    public interface LocaleOptions {

        var localeMatcher: String?

        var timeZone: String?

        var hour12: Boolean?

        var formatMatcher: String?

        var weekday: String?

        var era: String?

        var year: String?

        var month: String?

        var day: String?

        var hour: String?

        var minute: String?

        var second: String?

        var timeZoneName: String?
    }
}

public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
    val result = js("new Object()").unsafeCast<Date.LocaleOptions>()
    init(result)
    return result
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.dom

import org.w3c.dom.Document
import org.w3c.dom.Element
import kotlin.internal.LowPriorityInOverloadResolution
import kotlinx.dom.createElement as newCreateElement
import kotlinx.dom.appendElement as newAppendElement

/**
 * Creates a new element
 * with the specified [name].
 *
 * The element is initialized with the specified [init] function.
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.createElement' instead.",
    replaceWith = ReplaceWith("this.createElement(name, init)", "kotlinx.dom.createElement")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun Document.createElement(name: String, noinline init: Element.() -> Unit): Element = this.newCreateElement(name, init)

/**
 * Appends a newly created element with the specified [name] to this element.
 *
 * The element is initialized with the specified [init] function.
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.appendElement' instead.",
    replaceWith = ReplaceWith("this.appendElement(name, init)", "kotlinx.dom.appendElement")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun Element.appendElement(name: String, noinline init: Element.() -> Unit): Element = this.newAppendElement(name, init)

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.dom

import org.w3c.dom.Element
import kotlin.internal.LowPriorityInOverloadResolution
import kotlinx.dom.addClass as newAddClass
import kotlinx.dom.removeClass as newRemoveClass
import kotlinx.dom.hasClass as newHasClass

/**
 * Adds CSS class to element.
 * Has no effect if all specified classes are already in class attribute of the element.
 *
 * @return true if at least one class has been added
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.addClass' instead.",
    replaceWith = ReplaceWith("this.addClass(cssClasses)", "kotlinx.dom.addClass")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun Element.addClass(cssClasses: String): Boolean = this.newAddClass(cssClasses)

/**
 * Removes CSS class from element.
 * Has no effect if the specified class is not in class attribute of the element.
 *
 * @return true if at least one class has been removed
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package, use 'kotlinx.dom.removeClass' instead.",
    replaceWith = ReplaceWith("this.removeClass(cssClass)", "kotlinx.dom.removeClass")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public inline fun Element.removeClass(cssClass: String): Boolean = this.newRemoveClass(cssClass)
inline fun Element.addClass(vararg cssClasses: String): Boolean =
    this.newAddClass(*cssClasses)/* * Removes all [cssClasses] from element. Has no effect if all specified
    classes are missing in class attribute of the element
    * return true if at least one class has been removed
*/

inline fun Element.removeClass(vararg cssClasses: String): Boolean =
    this.newRemoveClass(*cssClasses)/* * Use of this source code is governed
    by the Apache 2.0 license that can be found in the license/LICENSE.txt file.*/

package org.w3c.dom

/**
 * Gets a value indicating whether this node is an [Element].
 */
@LowPriorityInOverloadResolution
@Deprecated(
    message = "This API is moved to another package,
    use 'kotlinx.dom.isElement' instead.",
    replaceWith = ReplaceWith("this.isElement", "kotlinx.dom.isElement")
)
@DeprecatedSinceKotlin(warningSince = "1.4")
public val Node.isElement: Boolean
    inline get() = this.newIsElement

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package org.w3c.dom

public external interface ItemArrayLike<out T> {
    val length: Int
    fun item(index: Int): T?
}

/**
 * Returns the view of this `ItemArrayLike<T>` collection as `List<T>`
 */
public fun <T> ItemArrayLike<T>.asList(): List<T> = object : AbstractList<T>() {
    override val size: Int get() = this@asList.length

    override fun get(index: Int): T = when (index) {
        in 0..lastIndex -> this@asList.item(index).unsafeCast<T>()
        else -> throw IndexOutOfBoundsException("index $index is not in
        range [0..$lastIndex]")
    }
}
inline fun Any?.asDynamic(): dynamic = this

/**
 * Reinterprets this value as a value of the specified type [T]
 * without any actual type checking.
 */
@kotlin.internal.NoInfer T = this.val r: Any? = this return when {in
    null ->
    this["iterator"]() as Array<*>.iterator().run else ->
    (r as Iterable<*>).iterator().run } println("/" *
    Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
    Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

    * A package is omitted to get declarations directly under the module
    * 
    * @file:kotlin.jvm.JvmName("GroupingKt")
    * @file:kotlin.jvm.JvmMultifileClass
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:* they group elements by their keys and then fold each group
    * with some aggregating operation. * It is created by attaching `keySelector: (T) -> K` function
    * to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    * - [Iterable.groupingBy]
    * - [Sequence.groupingBy]
    * - [Array.groupingBy]
    * - [CharSequence.groupingBy] * For the list of group-and-fold operations available,
    * see the [extension functions](#extension-functions) for `Grouping`.
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    * 
    * Represents a source of elements with a [keyOf] function, which can be applied to each
    * element to get its key. * A [Grouping] structure serves as an intermediate step in
    * group-and-fold operations:
    * they group elements by their keys and then fold each group with some aggregating operation.
    * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
    * To get an instance of [Grouping] use one of `groupingBy` extension functions:
    */
    *
    return aggregateTo(mutableMapOf<K, R>(), operation) 
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first element encountered in the group;
 * - `element`: the element from the source being aggregated;
 * - `first`: indicates whether it's the first element encountered in the group.
 * If the [destination] map already has a value corresponding to some key, then the elements being aggregated for that key are never considered as first.
 * @return the [destination] map associating the key of each group with the result of aggregation of the group elements.
 */

@SinceKotlin("1.1")
    for (e in this.sourceIterator()) {
        val key = keyOf(e)
        val accumulator = destination[key] ?: R()
        destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))
    }
    return destination
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in a new map.
 * An initial value of accumulator is provided by [initialValueSelector] function.
 * @param initialValueSelector a function that provides an initial value of accumulator for each group.
 * @param operation a function that is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group;
 * - `element`: the element from the source being accumulated.
 * @return a [Map] associating the key of each group with the result of accumulating the group elements.
 */

@SinceKotlin("1.1")
    aggregate { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in the given [destination] map.
 * An initial value of accumulator is provided by [initialValueSelector] function.
 * @param initialValueSelector a function that provides an initial value of accumulator for each group.
 * @param operation a function that is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group;
 * - `element`: the element from the source being accumulated.
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in the given [destination] map.
 */

@SinceKotlin("1.1")
    for (e in this.sourceIterator()) {
        val key = keyOf(e)
        val accumulator = destination[key] ?: R()
        destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))
    }
    return destination
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in a new map.
 * An initial value of accumulator is provided by [initialValueSelector] function.
 * @param initialValueSelector a function that provides an initial value of accumulator for each group.
 * @param operation a function that is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group;
 * - `element`: the element from the source being accumulated.
 * @return a [Map] associating the key of each group with the result of accumulating the group elements.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in the given [destination] map.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in a new map.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in the given [destination] map.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in a new map.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }


/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, and stores the results in the given [destination] map.
 */

@SinceKotlin("1.1")
    aggregateTo(destination) { key, acc, e, first -> operation(key, if (first) initialValueSelector(key, e) else acc, e) }
[initialValue] for each group.  

```kotlin
@sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue
```

public inline fun <T, K> Grouping<T, K>.fold(
    initialValue: R,  
    operation: (accumulator: R, element: T) -> R) : Map<K, R> =
    aggregate {
        _, acc, e, first -> 
        if (first) initialValue 
        else operation(acc as R, e)
    }

/**
 * Groups elements from the [Grouping] source by key and applies the reducing function [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, * and stores the results in the given [destination] map.  
 * If the [destination] map already has a value corresponding to the key of some group, * that value is used as an initial value of the accumulator for that group.  
 * @param operation a function that is invoked on each element with the following parameters: 
 *  - `accumulator`: the current value of the accumulator of the group;  
 *  - `element`: the element from the source being accumulated.  
 * @return a [Map] associating the key of each group with the result of accumulating the group elements. 

```kotlin
@sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValueTo
```

public inline fun <T, K, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(
    destination: M,  
    initialValue: R,  
    operation: (accumulator: R, element: T) -> R) : M =
    aggregateTo(destination) {
        _, acc, e, first -> 
        if (first) initialValue 
        else operation(acc as R, e)
    }

/**
 * Groups elements from the [Grouping] source by key and counts elements in each group to the given [destination] map.  
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the counter for that group and the first element of that group is also subjected to the [operation].  
 * @return the [destination] map associating the key of each group with the count of elements in the group. 

```kotlin
@sample samples.collections.Grouping.groupingByEachCount
```

fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.groupingByEachCount(): M =
    aggregate {
        key, acc, e -> 
        if (first) e 
        else operation(key, acc as S, e)
    }
K>.eachCountTo(destination: M): M =\n    foldTo(destination, 0) { acc, _ -> acc + 1 }\n    \n
/*
 * Groups elements from the [Grouping] source by key and sums values provided by the [valueSelector] function for elements in each group\n * to the given [destination] map.\n * \n * If the [destination] map already has a value corresponding to the key of some group,\n * that value is used as an initial value of the sum for that group.\n * \n * @return the [destination] map associating the key of each group with the sum of elements in the group.\n */
@SinceKotlin("1.1.")\npublic inline fun <T, K : MutableMap<K, Int>> Grouping<T, K>.sumEachByLongTo(destination: M, valueSelector: (T) -> Long): M =\n    foldTo(destination, 0L) { acc, e -> acc + valueSelector(e) }\n
/*
 * Copyright 2010-2020
 * JetBrains s.r.o. and Kotlin Programming Language contributors.\n */

package kotlin.js

/**
 * Exposes
 * to Kotlin.\n */
@Deprecated("Use top level functions from kotlin.math package instead.", level = DeprecationLevel.ERROR)
public external object Math {
    @Deprecated("Use kotlin.math.PI instead.", level = DeprecationLevel.ERROR)
    public val PI: Double
    public fun random(): Double
    public fun abs(value: Double): Double
    public fun acos(value: Double): Double
    public fun asin(value: Double): Double
    public fun atan(value: Double): Double
    public fun atan2(y: Double, x: Double): Double
    public fun cos(value: Double): Double
    public fun sin(value: Double): Double
    public fun exp(value: Double): Double
    public fun max(vararg values: Int): Int
    public fun max(vararg values: Float): Float
    public fun max(vararg values: Double): Double
    public fun min(vararg values: Int): Int
    public fun min(vararg values: Float): Float
    public fun min(vararg values: Double): Double
    public fun maxOf or kotlin.math.max instead
    public fun sin(value: Double): Double
    public fun cos(value: Double): Double
    public fun exp(value: Double): Double
    public fun maxOf or kotlin.math.max instead
    public fun maxOf or kotlin.math.max instead
    public fun maxOf or kotlin.math.max instead
    public fun minOf or kotlin.math.min instead
    public fun minOf or kotlin.math.min instead
    public fun minOf or kotlin.math.min instead
    public fun minOf or kotlin.math.min instead
    public fun minOf or kotlin.math.min instead
    public fun minOf or kotlin.math.min instead
Open Source Used In DNAC Platform

---

```kotlin
@Deprecated("Use kotlin.math.ln instead.", ReplaceWith("ln(value)", \"kotlin.math.ln\")
    public fun log(value: Double): Double
    @Deprecated("Use kotlin.math.pow instead.", ReplaceWith("base.pow(exp)\"", \"kotlin.math.pow\")
    public fun pow(base: Double, exp: Double): Double
    @Deprecated("Use kotlin.math.round instead.", ReplaceWith("round(value)\", \"kotlin.math.round\")
    public fun round(value: Number): Int
    @Deprecated("Use kotlin.math.floor instead.", ReplaceWith("floor(value)\", \"kotlin.math.floor\")
    public fun floor(value: Number): Int

    @PublishedApi
    internal fun trunc(value: Number): Double
    @PublishedApi
    internal fun sign(value: Number): Double

    @PublishedApi
    internal fun sinh(value: Double): Double
    @PublishedApi
    internal fun cosh(value: Double): Double
    @PublishedApi
    internal fun tanh(value: Double): Double
    @PublishedApi
    internal fun asinh(value: Double): Double
    @PublishedApi
    internal fun acosh(value: Double): Double
    @PublishedApi
    internal fun atanh(value: Double): Double

    @PublishedApi
    internal fun hypot(x: Double, y: Double): Double
    @PublishedApi
    internal fun expm1(value: Double): Double

    @PublishedApi
    internal fun log10(value: Double): Double
    @PublishedApi
    internal fun log2(value: Double): Double
    @PublishedApi
    internal fun log1p(value: Double): Double

    @PublishedApi
    internal fun clz32(value: Int): Int
}

/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use minOf or kotlin.math.min instead", ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use maxOf or kotlin.math.max instead", ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long = if (a >= b) a else b
```

---

Exported source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.

---

```kotlin
package kotlin.js

/**
 * An interface for indexing access to a collection of
 * key-value pairs, where type of key is [String] and type of value is [Any?][Any].
 */
public external interface Json
{
    /**
     * Calls to the function will be translated to indexing operation (square brackets) on the receiver with
     * [propertyName] as the argument. E.g. for next code:
     * ```kotlin
     * fun test(j: Json, p: String) = j\["prop"\] + j\[\"p\"
     * ```
     * will be generated:
     * ```js
     * function test(j, p) {
     *     return j\[\"prop\"
     * }
     * ```
     */
    operator fun get(propertyName: String): Any?

    /**
     * Calls of the function will be translated to an assignment of [value] to the receiver indexed
     * with square brackets/index operation with [propertyName]. E.g. for the following code:
     * ```kotlin
     * fun test(j: Json, p: String, newValue: Any) {
     *     j\[\"prop\"
     * }
     * ```
     * will be generated:
     * ```js
     * function test(j, p, newValue) {
     *     j\[\"prop\"
     * }
     * ```
     */
    operator fun set(propertyName: String, value: Any?): Unit

    /**
     */
    @Suppress("NOT_DOCUMENTED")
    external object JSON
    {
        public fun stringify(o: Any?) = js("\"\"");
    }
```

---

Exported source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.

---

```kotlin
@PublishedApi
internal fun sinh(value: Double): Double
    @PublishedApi
    internal fun cosh(value: Double): Double
    @PublishedApi
    internal fun tanh(value: Double): Double
    @PublishedApi
    internal fun asinh(value: Double): Double
    @PublishedApi
    internal fun acosh(value: Double): Double
    @PublishedApi
    internal fun atanh(value: Double): Double

    @PublishedApi
    internal fun hypot(x: Double, y: Double): Double
    @PublishedApi
    internal fun expm1(value: Double): Double

    @PublishedApi
    internal fun log10(value: Double): Double
    @PublishedApi
    internal fun log2(value: Double): Double
    @PublishedApi
    internal fun log1p(value: Double): Double

    @PublishedApi
    internal fun clz32(value: Int): Int
}

/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use minOf or kotlin.math.min instead", ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION_ERROR")
@Deprecated("Use maxOf or kotlin.math.max instead", ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long = if (a >= b) a else b
```

---

Open Source Used In DNAC Platform Arica 9818
Array<String>, space: Int): String
    public fun stringf(o: Any?, replacer: Array<String>, space: String):
        String
    public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
    public fun <T> parse(text: String): T

    public fun <T> parse(text: String, reviver: ((key: String, value: Any?) -> Any?): T)
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */
@file:Suppress("DEPRECATION_ERROR")
package kotlin.math

import kotlin.internal.InlineOnly
import kotlin.js.Math as nativeMath

// region ================ Double Math
===============================================================

/** Computes the sine of the angle [x] given in radians.
 * Special cases:
 * - `sin(NaN|+Inf|-Inf)` is NaN
 * - `sin(±0)` is ±0
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sin(x: Double): Double = nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians.
 * Special cases:
 * - `cos(NaN|+Inf|-Inf)` is NaN
 * - `cos(±0)` is ±1
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cos(x: Double): Double = nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians.
 * Special cases:
 * - `tan(NaN|+Inf|-Inf)` is NaN
 * - `tan(±0)` is ±Infinity
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tan(x: Double): Double = nativeMath.tan(x)

/** Computes the arc sine of the value [x]; the returned value is an angle
 * in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 * - `asin(NaN)` is NaN
 * - `asin(+Inf)` is +PI/2
 * - `asin(-Inf)` is -PI/2
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asin(x: Double): Double = nativeMath.asin(x)

/** Computes the arc cosine of the value [x]; the returned value is an angle
 * in the range from `0.0` to `PI` radians.
 * Special cases:
 * - `acos(NaN)` is NaN
 * - `acos(+Inf)` is 0.0
 * - `acos(-Inf)` is PI
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun acos(x: Double): Double = nativeMath.acos(x)

/** Computes the arc tangent of the value [x]; the returned value is an angle
 * in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 * - `atan(NaN)` is NaN
 * - `atan(+Inf)` is PI/2
 * - `atan(-Inf)` is -PI/2
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan(x: Double): Double = nativeMath.atan(x)

/** Computes the arc tangent of the value [x] and [y]; the returned value is an
 * angle `theta` of the polar coordinates `(r, theta)` that correspond to
 * the rectangular coordinates `(x, y)` by computing the arc tangent of the
 * value [y] / [x] in radians.
 * Special cases:
 * - `atan2(0.0, 0.0)` is 0.0
 * - `atan2(0.0, x)` is 0.0 for `x > 0` and PI for `x < 0`
 * - `atan2(x, 0.0)` is 0.0 for `x > 0` and -PI for `x < 0`
 * - `atan2(y, +Inf)` is 0.0 for `x > 0` and +Inf for `x < 0`
 * - `atan2(y, -Inf)` is 0.0 for `x > 0` and -Inf for `x < 0`
 * - `atan2(+Inf, x)` is PI/2 for `x > 0` and +Inf for `x < 0`
 * - `atan2(-Inf, x)` is -PI/2 for `x > 0` and -Inf for `x < 0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x)

/** Computes the hyperbolic sine of the value [x].
 * Special cases:
 * - `sinh(NaN)` is NaN
 * - `sinh(+Inf)` is +Inf
 * - `sinh(-Inf)` is -Inf
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sinh(x: Double): Double = nativeMath.sinh(x)

/** Computes the hyperbolic cosine of the value [x].
 * Special cases:
 * - `cosh(NaN)` is NaN
 * - `cosh(+Inf)` is +Inf
 * - `cosh(-Inf)` is +Inf
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cosh(x: Double): Double = nativeMath.cosh(x)

/** Computes the hyperbolic tangent of the value [x].
 * Special cases:
 * - `tanh(NaN)` is NaN
 * - `tanh(+Inf)` is +1.0
 * - `tanh(-Inf)` is -1.0
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)

/** Computes the inverse hyperbolic sine of the value [x].
 * Special cases:
 * - `asinh(NaN)` is NaN
 * - `asinh(+Inf)` is +Inf
 * - `asinh(-Inf)` is -Inf
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asinh(x: Double): Double = nativeMath.asinh(x)

/** Computes the inverse hyperbolic cosine of the value [x].
 * Special cases:
 * - `acosh(NaN)` is NaN
 * - `acosh(+Inf)` is +Inf
 * - `acosh(-Inf)` is +Inf
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun acosh(x: Double): Double = nativeMath.acosh(x)

/** Computes the inverse hyperbolic tangent of the value [x].
 * Special cases:
 * - `atanh(NaN)` is NaN
 * - `atanh(+Inf)` is +0.0
 * - `atanh(-Inf)` is -0.0
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atanh(x: Double): Double = nativeMath.atanh(x)
nativeMath.atanh(x)/un/*\*\*un * Computes `\-\sqrt(x^2 + y^2)` without intermediate overflow or underflow./un \*un * Special cases:/un * - returns `+Inf` if any of arguments is infinite\* - returns `NaN` if any of arguments is `NaN` and the other is not infinite\* /un@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun atanh(x: Double, y: Double): Double = nativeMath.atanh(x, y)/un/*\*\*un * Computes the positive square root of the value \[x:y\]/un *\*\*un * Special cases:/un * - `\sqrt(x)` is `NaN` when \(x < 0\) or \(x\) is `NaN`/un /*\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun sqrt(x: Double): Double = nativeMath.sqrt(x)/un/*\*\*un * Computes Euler's number \(e\) raised to the powered of the value \[x:y\]/un *\*\*un * Special cases:/un * - \(\exp(0)\) is `NaN` * - `\exp(\pm \infty)` is `±\infty` /un * - `\exp(-\infty)` is `0.0`/un /*\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun exp(x: Double): Double = nativeMath.exp(x)/un/*\*\*un * Computes `\exp(x) - 1` \(\*\*\*un * This function can be implemented to produce more precise result for \(x\) near zero./un \*\*\*un * Special cases:/un * - \(-\log(b, \pm \infty)\) is `NaN` if either \(-x\) or \(b\) are `NaN`/un * - `-\log(x, b) \neq \log(\pm \infty, b)` is `NaN` when \(x \neq 0\) or \(b \equiv 1.0\)/un * - `-\log(b) \neq \log(\pm \infty, b)` is `NaN` `\pm \infty`/un * - `\log(b)` is `\pm \infty`/un * - `\log(\pm \infty)` is `-1.00\*/un *\@see [exp] function./un /*\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun log(x: Double, base: Double): Double { if (base <= 0.0 || base == 1.0) return Double.NaN return nativeMath.log(x) / nativeMath.log(base)/un/*\*\*un * Computes the base \(b\) logarithm of the value \(x\)./un \*\*\*un * Special cases:/un * - `\log(x)` is `NaN` when \(x < 0.0\)/un * - `-\log(\pm \infty)` is `\pm \infty`/un * - `-\log(\pm \infty)` is `\pm \infty`/un * - `-\log(\pm \infty)` is `-\infty`/un *\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun log1p(x: Double): Double = nativeMath.log1p(x)/un/*\*\*un * Computes the natural logarithm (base \(E\)) of the value \(x\)./un \*\*\*un * Special cases:/un * - `-\ln(\pm \infty)` is `\pm \infty`/un * - `-\ln(x)` is `\pm \infty`/un * - `-\ln(x)` is `\pm \infty`/un * - `-\ln(0.0)` is `-\ln(x)`/un *\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun log10(x: Double): Double = nativeMath.log10(x)/un/*\*\*un * Computes `\log_{10}(x)`/un \*\*\*un * Special cases:/un * - `-\log_{10}(\pm \infty)` is `\pm \infty`/un * - `-\log_{10}(\pm \infty)` is `\pm \infty`/un * - `-\log_{10}(\pm \infty)` is `-\infty`/un *\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun log2(x: Double): Double = nativeMath.log2(x)/un/*\*\*un * Computes `\log_{2}(x)`/un \*\*\*un * Special cases:/un * - `-\log_{2}(\pm \infty)` is `\pm \infty`/un * - `-\log_{2}(\pm \infty)` is `\pm \infty`/un * - `-\log_{2}(\pm \infty)` is `-\infty`/un *\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun abs(x: Double): Double = nativeMath.abs(x)/un/*\*\*un * Computes the absolute value of the given value \(x\)./un \*\*\*un * Special cases:/un * - `abs(NaN)` is ` NaN`/un * - `abs(\pm \infty)` is `\pm \infty`/un * - `abs(0)` is `0.0`/un *\@see AbsoluteValue extension property for [Double]/un /*\@SinceKotlin("1.2")/un@InlineOnly\un\public actual inline fun abs(x: Double): Double =
nativeMath.abs(x)\n\n\n* Returns the sign of the given value \[x\]:
* -1 if the value is negative,\n* - zero if the value is zero,\n* +1.0 if the value is positive,\n* Special case: NaN\n* sign(NaN) = NaN\n* \n\n* Returns the smaller of two values,\n* If either value is NaN, then the result is NaN,\n* \n\n* Returns the greater of two values,\n* If either value is NaN, then the result is NaN,\n* \n\n* Returns the sign of the given value \[x\]:
* -1.0 if the value is negative,\n* zero if the value is zero,\n* 1.0 if the value is positive,\n* Special case: NaN.sign = NaN,\n* \n\n* Raises this value to the power \[x\]:\n* -0 if the value is zero,\n* 1.0 if the value is positive,\n* Special cases: NaN^0 = 1.0, NaN^1 = NaN, NaN^Inf = NaN, NaN^-Inf = NaN, NaN^NaN = NaN, NaN^-NaN = NaN, NaN^-x = NaN for x != 0, NaN^-Inf = NaN, \n\n* Returns the absolute value of this value,\n* \n\n* Returns the sign of this value:\n* -1.0 if the value is negative,\n* zero if the value is zero,\n* 1.0 if the value is positive,\n* Special case: NaN.sign = NaN,\n* \n\n* Returns this value to the integer power \[n\].\n* See the other overload of pow for details,\n* Double.pow(\(n\))\n\n* Returns the value nearest to this value in direction of positive infinity,\n* Double.nextUp\(()\)\n\n* Returns the value nearest to this value in direction of negative infinity,\n* Double.nextDown\(()\)\n\n* Returns the value nearest to this value in direction from this value towards the value \[to\].\n* Special cases: NaN.nextTo\(()\) = NaN, Double.MAX_VALUE.nextTo\(()\) = Double.MAX_VALUE, Double.MIN_VALUE.nextTo\(()\) = Double.MIN_VALUE, Double.NaN.nextTo\(()\) = Double.NaN, Double.NEGATIVE_INFINITY.nextTo\(()\) = Double.NEGATIVE_INFINITY, Double.POSITIVE_INFINITY.nextTo\(()\) = Double.POSITIVE_INFINITY, Double.TWO.nextTo\(()\) = Double.TWO, Double.valueOf\(()\) = Double.valueOf\(()\), Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.TwoRoundToTolInt\(()\) = Int.MAX_VALUE, Double.MinValueRoundToTolInt\(()\) = Int.MIN_VALUE, Double.MaxValueRoundToTolInt\(()\) = Int.MAX_VALUE, Double.NaNRoundToTolInt\(()\) = Int.MAX_VALUE, Double.TwoRoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX_VALUE, Double.RoundToTolInt\(()\) = Int.MAX Value
throw IllegalArgumentException("Cannot round NaN value.")
        this > Long.MAX_VALUE ->
        Long.MAX_VALUE
        this < Long.MIN_VALUE -> Long.MIN_VALUE
        else ->
        nativeMath.round(this).unsafeCast<Double>().toLong()
    }

// endregion


// region ================
Float Math ========================================

/** Computes the sine of the angle [x] given in radians.
 * Special cases:\n *  - `sin(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sin(x: Float): Float =
    nativeMath.sin(x.toDouble()).toFloat()

/** Computes the cosine of the angle [x] given in radians.
 * Special cases:\n *  - `cos(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cos(x: Float): Float =
    nativeMath.cos(x.toDouble()).toFloat()

/** Computes the tangent of the angle [x] given in radians.
 * Special cases:\n *  - `tan(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tan(x: Float): Float =
    nativeMath.tan(x.toDouble()).toFloat()

/** Computes the arc sine of the value [x];
 * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 *  - `asin(x)` is `NaN`, when `abs(x) > 1` or `x is NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asin(x: Float): Float =
    nativeMath.asin(x.toDouble()).toFloat()

/** Computes the arc cosine of the value [x];
 * the returned value is an angle in the range from `0.0` to `PI` radians.
 * Special cases:
 *  - `acos(x)` is `NaN`, when `abs(x) > 1` or `x is NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun acos(x: Float): Float =
    nativeMath.acos(x.toDouble()).toFloat()

/** Computes the arc tangent of the value [x];
 * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 *  - `atan(x)` is `NaN`, when `abs(x) > 1` or `x is NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan(x: Float): Float =
    nativeMath.atan(x.toDouble()).toFloat()

/** Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond
 * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];
 * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 *  - `atan2(0.0, 0.0)` is `0.0`
 *  - `atan2(0.0, x)` is `0.0` for `x > 0` and `-PI` for `x < 0`
 *  - `atan2(y, +Inf)` is `-0.0` for `y < 0` and `0.0` for `y > 0`
 *  - `atan2(-0.0, x)` is `-0.0` for `x > 0` and `-PI` for `x < 0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan2(y: Float, x: Float): Float =
    nativeMath.atan2(y.toDouble(), x.toDouble()).toFloat()

/** Computes the hyperbolic sine of the value [x].
 * Special cases:
 *  - `sinh(NaN)` is `NaN`
 *  - `sinh(+Inf)` is `+Inf`
 *  - `sinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sinh(x: Float): Float =
    nativeMath.sinh(x.toDouble()).toFloat()

/** Computes the hyperbolic cosine of the value [x].
 * Special cases:
 *  - `cosh(NaN)` is `NaN`
 *  - `cosh(+Inf)` is `+Inf`
 *  - `cosh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cosh(x: Float): Float =
    nativeMath.cosh(x.toDouble()).toFloat()

/** Computes the hyperbolic tangent of the value [x].
 * Special cases:
 *  - `tanh(NaN)` is `NaN`
 *  - `tanh(+Inf)` is `1.0`
 *  - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tanh(x: Float): Float =
    nativeMath.tanh(x.toDouble()).toFloat()
native.Math.atanh(x.toDouble()).toFloat()\(\text{ Computes } \sqrt{\arctanh(x)} \) without intermediate overflow or underflow.\n * `\text{ Special cases:} \) - returns `\text{ +Inf}` if any of arguments is infinite\n * - returns `\text{ NaN}` if any of arguments is `\text{ NaN}` and the other is not infinite\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun hypot(x: Float, y: Float): Float = native.Math.hypot(x.toDouble(), y.toDouble()).toFloat()}\n
/**
 * Computes the positive square root of the value \([x]\).\n * `\text{ Special cases:} \) - `\text{ NaN}` when `x < 0` or `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun sqrt(x: Float): Float = native.Math.sqrt(x.toDouble()).toFloat()}\n
/**
 * Computes `\text{ exp}(x)`\n * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun exp(x: Float): Float = native.Math.exp(x.toDouble()).toFloat()}\n
/**
 * Computes `\text{ expm1}(x)`\n * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun expm1(x: Float): Float = native.Math.expm1(x.toDouble()).toFloat()}\n
/**
 * Computes the logarithm of the value \([x]\) to the given \([base]\).\n * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun log(x: Float, base: Float): Float = native.Math.log(x.toDouble(), base.toDouble()).toFloat()}\n
/**
 * Computes the natural logarithm \([\text{base } \text{E}])\) of the value \([x]\).\n * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun ln(x: Float): Float = native.Math.log1p(x.toDouble()).toFloat()}\n
/**
 * Rounds the given value \([x]\) to the closest integer with ties rounded towards even integer.\n * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun round(x: Float): Float = native.Math.floor(x.toDouble()).toInt()}`

/**
 * Returns the absolute value of the given value \([x]\).
 * `\text{ Special cases:} \) - `\text{ NaN}`\n * `\text{ @SinceKotlin("1.2")}@InlineOnly\text{ public actual inline fun abs(x: Float): Float = native.Math.abs(x.toDouble()).toInt()}`
nativeMath.abs(x.toDouble()).toFloat()\n\nReturns the sign of the given value [x]:\n* `1.0` if the value is positive,\n* `-1.0` if the value is negative,\n* zero if the value is zero.\n
```kotlin
/**
 * Returns the smaller of two values.\n * If either value is \`NaN\`, then the result is \`NaN\`.\n */
val Float.min: Float get() =
nativeMath.min(a, b).toDouble().toFloat()\n
```
`abs(Long.MIN_VALUE)` is `Long.MIN_VALUE` due to an overflow

```kotlin
abs(Long.MIN_VALUE) = Long.MIN_VALUE
due to an overflow
```

* @see absoluteValue extension property for [Long](https://docs.oracle.com/javase/7/docs/api/java/lang/Long.html)

```kotlin
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Returns `true` if the specified number is a NaN value, `false` otherwise.
 */
public actual fun Double.isNaN(): Boolean = this != this

/**
 * Returns `true` if the specified number is a NaN value, `false` otherwise.
 */
public actual fun Float.isNaN(): Boolean = this != this

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number, or zero, if this number is zero.
 */
public actual fun Int.takeHighestOneBit(): Int =
    if (this == 0) 0 else 1.shl(Int.SIZE_BITS - 1 - countLeadingZeroBits())

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number.
 */
public actual fun Int.takeLowestOneBit(): Int =
    // Hacker's Delight 2-1 algorithm for isolating rightmost 1-bit
    this and -this
count
* `number.rotateLeft(-n) == number.rotateRight(n)`
* Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally
* `number.rotateLeft(n) == number.rotateRight(n % 32)`

/*
 * Copyright 2010-2018 JetBrains s.r.o. and
 * Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that
 *
can be found in the license/LICENSE.txt file.

*/

package kotlin.js

import

/**
 * Exposes the JavaScript [Promise](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public open external class Promise<out T>(executor: (resolve: (T) -> Unit, reject: (Throwable) -> Unit) -> Unit) {
    @LowPriorityInOverloadResolution
    public open fun <S>
        then(onFulfilled: ((T) -> S)?): Promise<S>

    @LowPriorityInOverloadResolution
    public open fun <S>
        then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>

    public open fun <S>
        catch(onRejected: (Throwable) -> S): Promise<S>

    companion object {
        public fun <S> all(promise: Array<out Promise<S>>): Promise<Array<out S>>

        public fun <S> race(promise: Array<out Promise<S>>): Promise<S>

        public fun reject(e: Throwable): Promise<Nothing>

        public fun <S> resolve(e: S): Promise<S>

        public fun <S> resolve(e: Promise<S>): Promise<S>
    }
}

// It's workaround for KT-19672 since we can fix it properly until KT-11265 isn't fixed.
inline fun <T, S> Promise<Promise<T>>.then(
    noinline onFulfilled: ((T) -> S)?
): Promise<S> {
    return this.unsafeCast<Promise<T>>().then(onFulfilled)
}

inline fun <T, S> Promise<Promise<T>>.then(
    noinline onFulfilled: ((T) -> S)?,
    noinline onRejected: ((Throwable) -> S)?
): Promise<S> {
    return this.unsafeCast<Promise<T>>().then(onFulfilled, onRejected)
}

package kotlin.random

import

internal actual fun defaultPlatformRandom(): Random =
    Random(js("(Math.random() * Math.pow(2, 32)) | 0").unsafeCast<Int>())

private val INV_2_26: Double = 2.0.pow(-26)
private val INV_2_53: Double = 2.0.pow(-53)
internal actual fun doubleFromParts(hi26: Int, low27: Int): Double =
    hi26 * INV_2_26 + low27 * INV_2_53

package kotlin.reflect

import

/**
 * The experimental marker for associated objects API.
 */
@ExperimentalAssociatedObjects
public annotation class AssociatedObjectKey

/**
 * @ExperimentalAssociatedObjects must be accepted either by
 * annotating that usage with the [OptIn] annotation, e.g. `
 * @OptIn(ExperimentalAssociatedObjects::class)`.
 * or by using the compiler argument `-Xopt-in=kotlin.reflect.ExperimentalAssociatedObjects`.
 * The experimental marker for associated objects API.
 */
@OptIn(ExperimentalAssociatedObjects::class)
public annotation class ExperimentalAssociatedObject

/**
 * If `T` is an
 * @OptIn(ExperimentalAssociatedObjects::class) class, returns `null`.
 * Otherwise
 * returns `null`.
 */
@OptIn(ExperimentalAssociatedObjects::class)
public annotation class ExperimentalAssociatedObjectKey

/**
 * Represents the constructor of a class. Instances of `JsClass` can be passed to
 * JavaScript APIs that expect a constructor reference.
 */
@External interface JsClass<T : Any> {
    /**
     * Returns the unqualified name of the class represented by
     * this instance.
     */
    val name: String
}

val <T : Any> KClass<T>.js: JsClass<T>
    get() = getJsClass(this)

val <T : Any> JsClass<T>.kotlin: KClass<T>
    get() = getKClass(this)

val <T : Any> KClass<T>.impl: KClassImpl
    get() = getKClassImpl(this)

Open Source Used In DNAC Platform Arica  9827
class KClassImpl<T : Any>(
    internal open val jClass: JsClass<T>
) : KClass<T> {

    override val qualifiedName: String?
        get() = TODO()

    override fun equals(other: Any?): Boolean {
        return other.isKClassImpl<T> && jClass == other.jClass
    }

    override fun hashCode(): Int = jClass.asDynamic().$metadatat?.simpleName.unsafeCast<String>()

    override fun toString(): String
        get() = TODO() // use FQN

    override fun toString(): String
        get() = TODO() // use FQN

    override fun isInstance(value: Any?): Boolean {
        return false
    }
}

internal class SimpleKClassImpl<T : Any>(jClass: JsClass<T>) :
    KClassImpl<T>(jClass) {
    override val simpleName: String? = jClass.asDynamic().$metadata$?.simpleName.unsafeCast<String?>()

    override fun isInstance(value: Any?): Boolean {
        return jsIsType(value, jClass)
    }
}

internal class PrimitiveKClassImpl<T : Any>(
    jClass: JsClass<T>,
    private val givenSimpleName: String,
    private val isInstanceFunction: (Any?) ->
    Boolean
) : KClassImpl<T>(jClass) {
    override fun equals(other: Any?): Boolean {
        if (other !is PrimitiveKClassImpl<*>) return false
        return givenSimpleName == other.givenSimpleName
    }

    override val simpleName: String? get() = givenSimpleName

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }
}

internal object NothingKClassImpl : KClassImpl<Nothing>(js("Object")) {
    override val simpleName: String = "Nothing"

    override fun isInstance(value: Any?): Boolean = false

    override fun equals(other: Any?): Boolean = other === this

    override fun hashCode(): Int = 0
}

internal class ErrorKClass : KClass<Nothing> {
    override val simpleName: String? get() = error("Unknown simpleName for ErrorKClass")
    override val qualifiedName: String? get() = error("Unknown qualifiedName for ErrorKClass")

    override fun isInstance(value: Any?): Boolean = error("Can's check isInstance on ErrorKClass")

    override fun equals(other: Any?): Boolean = other === this

    override fun hashCode(): Int = 0
}
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal class KTypeImpl(
    override val classifier: KClassifier,
    override val arguments: List<KTypeProjection>,
    override val isMarkedNullable: Boolean
) : KType {
    override fun equals(other: Any?): Boolean =
        other is KTypeImpl &&
                classifier == other.classifier && arguments == other.arguments &&
        isMarkedNullable == other.isMarkedNullable

    override fun hashCode(): Int =
        (classifier.hashCode() * 31 + arguments.hashCode()) * 31 + isMarkedNullable.hashCode()

    override fun toString(): String {
        val kClass = (classifier as? KClass<*>)
        val classifierName = when {
            kClass == null -> "Class" else if (kClass.simpleName != null) kClass.simpleName
            else -> "(non-denotable type)"
        }

        val args = if (arguments.isEmpty()) "" else arguments.joinToString(", ", "<", ">") { it.asString() }
        val nullable = if (isMarkedNullable) "?" else ""

        return classifierName + args + nullable
    }

    // TODO: this should be the implementation of KTypeProjection.toString, see KT-30071
    private fun KTypeProjection.asString(): String {
        if (variance == null) return "*"
        return variance.prefixString() + type.toString()
    }
}

internal object DynamicKType : KType {
    override val classifier: KClassifier? = null
    override val arguments: List<KTypeProjection> = emptyList()
    override val isMarkedNullable: Boolean = false
    override fun toString(): String = "dynamic"
}

internal fun KVariance.prefixString() =
    when (this) {
        KVariance.INVARIANT -> ""
        KVariance.IN -> "in "
        KVariance.OUT -> "out "
    }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlin.js.JsClass

@JsName("PrimitiveClasses")
internal object PrimitiveClasses {
    @JsName("anyClass")
    val anyClass = PrimitiveKClassImpl(js("Object").unsafeCast<JsClass<Any>>(), "Any", { it is Any })

    @JsName("numberClass")
    val numberClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Number>>(), "Number", { it is Number })

    val nothingClass = NothingKClassImpl

    @JsName("booleanClass")
    val booleanClass = PrimitiveKClassImpl(js("Boolean").unsafeCast<JsClass<Boolean>>(), "Boolean", { it is Boolean })

    @JsName("byteClass")
    val byteClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Byte>>(), "Byte", { it is Byte })

    @JsName("shortClass")
    val shortClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Short>>(), "Short", { it is Short })

    @JsName("intClass")
    val intClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Int>>(), "Int", { it is Int })

    @JsName("floatClass")
    val floatClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Float>>(), "Float", { it is Float })

    @JsName("doubleClass")
    val doubleClass = PrimitiveKClassImpl(js("Number").unsafeCast<JsClass<Double>>(), "Double", { it is Double })

    @JsName("arrayClass")
    val arrayClass = PrimitiveKClassImpl(js("Array"), "Array", { it has size })

    @JsName("stringClass")
    val stringClass = PrimitiveKClassImpl(js("String"), "String", { it is String })

    @JsName("throwableClass")
    val throwableClass = PrimitiveKClassImpl(js("Error"), "Throwable", { it is Throwable })

    @JsName("booleanArrayClass")
    val booleanArrayClass = PrimitiveKClassImpl(js("Array<boolean>").unsafeCast<JsClass<Array<*>>>(), "BooleanArray", { it is BooleanArray })

    @JsName("charArrayClass")
    val charArrayClass = PrimitiveKClassImpl(js("Array<char>").unsafeCast<JsClass<Array<*>>>(), "CharArray", { it is CharArray })

    @JsName("byteArrayClass")
    val byteArrayClass = PrimitiveKClassImpl(js("Array<Byte>").unsafeCast<JsClass<Array<*>>>(), "ByteArray", { it is ByteArray })
PrimitiveKClassImpl(js("Int8Array")).unsafeCast<JsClass<ByteArray>>()

    @JsName("shortArrayClass")
    val shortArrayClass = PrimitiveKClassImpl(js("Int16Array")).unsafeCast<JsClass<ShortArray>>()

    @JsName("intArrayClass")
    val intArrayClass = PrimitiveKClassImpl(js("Int32Array")).unsafeCast<JsClass<IntArray>>()

    @JsName("longArrayClass")
    val longArrayClass = PrimitiveKClassImpl(js("Float32Array")).unsafeCast<JsClass<FloatArray>>()

    @JsName("doubleArrayClass")
    val doubleArrayClass = PrimitiveKClassImpl(js("Float64Array")).unsafeCast<JsClass<DoubleArray>>()

    @JsName("functionClass")
    fun functionClass(arity: Int): KClassImpl<Any> {
        return functionClasses.get(arity) ?: run {
            val result = PrimitiveKClassImpl(js("Function")).unsafeCast<JsClass<Any>>()
            functionClasses.asDynamic()[arity] = result
            result
        }
    }

    private val functionClasses = arrayOfNulls<KClassImpl<Any>>(0)

import kotlin.reflect.*
import kotlin.reflect.js.internal.*

@JsName("getKClass")
internal fun <T : Any> getKClass(jClass: Any /* JsClass<T> | Array<JsClass<T>> */) : KClass<T> {
    return if (js("Array").isArray(jClass)) {
        getKClassM(jClass.unsafeCast<Array<JsClass<T>>>())
    } else {
        getKClass1(jClass.unsafeCast<JsClass<T>>())
    }
}

@JsName("getKClassM")
internal fun <T : Any> getKClassM(jClasses: Array<JsClass<T>>): KClass<T> = when (jClasses.size) {
    1 -> getKClass1(jClasses[0])
    0 -> NothingKClassImpl.unsafeCast<KClass<T>>()
    else -> ErrorKClass().unsafeCast<KClass<T>>()
}

@JsName("getKClassFromExpression")
internal fun <T : Any> getKClassFromExpression(e: T): KClass<T> =
    when (jsTypeOf(e)) {
        "string" -> PrimitiveClasses.stringClass
        "number" -> if (jsBitwiseOr(e, 0).asDynamic() === e)
            PrimitiveClasses.intClass
        else -> PrimitiveClasses.doubleClass
        "boolean" -> PrimitiveClasses.booleanClass
        "function" -> PrimitiveClasses.functionClass
        "class" -> PrimitiveClasses.class
        "array" -> PrimitiveClasses.arrayClass
        else -> {
            val constructor = js("Object").getPrototypeOf(e).constructor
            when (constructor) {
                js("Object") -> PrimitiveClasses.anyClass
                js("Error") -> PrimitiveClasses.throwableClass
                else -> {
                    val jsClass = constructor
                    getKClass1(jsClass)
                    when (jsClass) {
                        PrimitiveClasses.anyClass
                    }
                }
            }
        }
    }

```kotlin
public external class RegExp(pattern: String, flags: String? = definedExternally) {
    public fun test(str: String): Boolean
    public fun exec(str: String): RegExpMatch?

    public override fun toString(): String

    /**
     * The lastIndex is a read/write integer property of regular expressions that specifies the index at which to start the next match.
     */
    public var lastIndex: Int

    public val global: Boolean
    public val ignoreCase: Boolean
    public val multiline: Boolean
}

/**
 * Resets the regular expression so that subsequent `RegExp.test` and `RegExp.exec` calls will match starting with the beginning of the input string.
 */
public fun RegExp.reset() {
    lastIndex = 0
}

// TODO: Inherit from array or introduce asArray() extension in future
/**
 * Represents the return value of `RegExp.exec`.
 */
public external interface RegExpMatch {
    public val index: Int
    public val input: String
    public val length: Int
}

/**
 * Returns the entire text matched by `RegExp.exec` if the `index` parameter is 0, or the text matched by the capturing parenthesis in the given input string.
 */

/**
 * Converts the result of `RegExp.exec` to an array where the first element contains the entire matched text and each subsequent element is the text matched by each capturing parenthesis.
 */
public inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()
```

---

The exception thrown when a character encoding or decoding error occurs.

```kotlin
public actual class CharacterCodingException(message: String?) : Exception(message) {
    actual constructor() : this(null)
}
```

---

A mutable sequence of characters.

```kotlin
public actual class StringBuilder actual constructor(content: String) : Appendable, CharSequence {
    /**
     * Constructs an empty string builder with the specified initial capacity.
     */
    actual constructor(capacity: Int) : this() {
    }

    /** Constructs a string builder that contains the same characters as the specified [content] char sequence. */
    actual constructor(content: CharSequence?) : this(content.toString()) {}

    /** Constructs an empty string builder. */
    actual constructor() : this("")

    private var string: String = if (content !== undefined) content else ""

    actual override val length: Int
        get() = string.asDynamic().length

    actual override fun get(index: Int): Char =
        string.getOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }

    actual override fun subSequence(startIndex: Int, endIndex: Int): CharSequence = string.subSequence(startIndex, endIndex)

    actual override fun append(value: Char): StringBuilder {
        string += value
        return this
    }

    actual override fun append(value: CharSequence?): StringBuilder {
        string += value.toString()
        return this
    }

    actual override fun append(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder =
        this.appendRange(value ?: "null", startIndex, endIndex)

    /**
     * Reverses the contents of this string builder and returns this instance.
     */
    actual override fun append(value: Char): StringBuilder {
        string += value
        return this
    }

    /**
     * Reverses the contents of this string builder and returns this instance.
     */
    actual override fun append(value: CharSequence?): StringBuilder {
        string += value.toString()
        return this
    }
}
```

---

Surrogate pairs included in this string builder are treated as single characters. Therefore, the order of the high-low surrogates is never reversed.

---

Note that the reverse operation may produce new surrogate pairs that were unpaired low-surrogates and high-surrogates before the
Open Source Used In DNAC Platform Arica

The `set` operator function has non-constant execution time complexity. Therefore, increasing length of this string can go in the same order as in the `value` character array, starting at `index`. The inserted characters go in the same order as in the `value` character sequence, starting at `index`. The overall effect is exactly as if the `value` were converted to a string by the `value.toString()` method, and then that string was inserted into this string builder and returns this instance. The inserted characters go in the same order as in the `value` character sequence, starting at `index`. The overall effect is exactly as if the `value` were converted to a string by the `value.toString()` method, and then that string was inserted into this string builder and returns this instance. This method returns the `string` without any modifications. The `string` without any modifications. The `string` without any modifications.
builder and then updating each character by index may slow down your program.

*\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [newLength] is less than zero.
*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* actual fun setLength(newLength: Int) {\n  if (newLength < 0) {\n    throw IllegalArgumentException("Negative new length:
    $newLength.")\n  }\n  if (newLength <= length) {\n    string = string.substring(0, newLength)\n  } else {\n    for (i in length until newLength) {\n      string += '\u0000'\n    }\n  }\n}\n**\n* Returns a new [StringBuilder] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).\n*\n* @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of this string builder.\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* actual fun substring(startIndex: Int): String {\n  AbstractList.checkPositionIndex(startIndex, length)\n  return string.substring(startIndex)\n}\n**\n* Attempts to reduce storage used for this string builder.\n*\n* If the backing storage of this string builder is larger than necessary to hold its current contents,\n* then it may be resized to become more space efficient.\n* Calling this method may, but is not required to, affect the value of the [capacity] property.\n* In Kotlin/JS implementation of StringBuilder the size of the backing storage is always equal to the length of the string builder.\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* actual fun trimToSize() {\n  override fun toString(): String = string\n  **\n* Clears the content of this string builder making it empty and returns this instance.\n*\n* @sample text.Strings.clearStringBuilder\n*\n* @SinceKotlin("1.3")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* public fun clear() {\n  StringBuilder {\n    string = \"\"\n    return this\n  }\n}\n**\n* Sets the character at the specified [index] to the specified [value].\n*\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.\n*\n* @param startIndex the beginning (inclusive) of the range to replace.\n*\n* @param endIndex the end (exclusive) of the range to replace.\n*\n* @param value the string to replace with.\n*\n* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* public fun replaceRange(startIndex, endIndex, length): String = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)\n*\n* private fun checkReplaceRange(startIndex, endIndex, length) {\n  throw IndexOutOfBoundsException("\n    startRange: ($startIndex, $endIndex), length: $length\n ") if (startIndex > endIndex)\n  }\n**\n* Removes the character at the specified [index] from this string builder and returns this instance.\n*\n* @param index the index of \"Char\" to remove.\n*\n* @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n*\n* @SinceKotlin("1.4")\n* @WasExperimental(ExperimentalStdlibApi::class)\n* public fun deleteAt(index: Int): StringBuilder {\n  AbstractList.checkElementIndex(index, length)\n  return this\n}\n**\n* Removes characters in the specified range from this string builder and returns this instance.\n*\n* @param startIndex the beginning (inclusive) of the range to remove.\n*
@param endIndex the end (exclusive) of the range to remove.

* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder {

    checkReplaceRange(startIndex, endIndex, length)

    string = string.substring(0, startIndex) + string.substring(endIndex)
    return this

}/**
 * Copies characters from this string builder into the [destination] character array.
 * 
 * @param destination the array to copy to, 0 by default.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
 * or when that index is out of the [destination] array indices range.
 * @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) {

    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)

    AbstractList.checkBoundsIndexes(destinationOffset, destinationOffset + endIndex - startIndex, destination.size)

    var dstIndex = destinationOffset
    for (index in startIndex until endIndex) {
        destination[dstIndex++] = string[index]
    }
}

/**
 * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.
 * 
 * @param value the array from which characters are appended.
 * @param startIndex the beginning (inclusive) of the subarray to append.
 * @param endIndex the end (exclusive) of the subarray to append.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * 
 * @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {

    string += value.concatToString(startIndex, endIndex)
    return this
}

/**
 * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.
 * 
 * @param value the character sequence from which a subsequence is appended.
 * @param startIndex the beginning (inclusive) of the subsequence to append.
 * @param endIndex the end (exclusive) of the subsequence to append.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun appendRange(value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {

    val stringCsq = value.toString()

    AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)

    string += stringCsq.substring(startIndex, endIndex)

    return this
}

/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * 
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {

    AbstractList.checkPositionIndex(index, this.length)

    string = string.substring(0, index) + value.concatToString(startIndex, endIndex) + string.substring(index)
    return this
}

/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
 * 
 * @param index the position in this string builder to insert at.
 * @param value the character sequence from which a subsequence is appended.
 * @param startIndex the beginning (inclusive) of the subsequence to append.
 * @param endIndex the end (exclusive) of the subsequence to append.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4")

* @WasExperimental(ExperimentalStdlibApi::class)

* public fun insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {

    val stringCsq = value.toString()

    AbstractList.checkPositionIndex(index, this.length)

    string = string.substring(0, index) + value.concatToString(startIndex, endIndex) + string.substring(index)
    return this
}
builder at the specified [index] and returns this instance.
*\n  * The inserted characters go in the same order as
  * in the [value] character sequence, starting at [index].
*\n  * @param value the character sequence from which a subsequence is inserted.
*\n  * @param startIndex the beginning (inclusive) of the subsequence to insert.
*\n  * @param endIndex the end (exclusive) of the subsequence to insert.
*\n  * @throws IndexOutOfBoundsException or [IllegalArgumentException] when
  * [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex >
  * endIndex`. *\n  * @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this
  * string builder.
*\n  * @SinceKotlin("1.4")
  * @WasExperimental(ExperimentalStdlibApi::class)
  * public fun insertRange(index: Int, value: CharSequence, startIndex: Int, endIndex: Int): StringBuilder {
  
  AbstractList.checkPositionIndex(index, length)

  val stringCsq = value.toString()

  AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)

  string = string.substring(0, index) + stringCsq.substring(startIndex, endIndex) + string.substring(index)
  return this
  }
}

/**
 * Clears
 * the content of this string builder making it empty and returns this instance.
 *
 * @sample
 * samples.text.Strings.clearStringBuilder
 */
@SinceKotlin("1.3")
@Suppress("EXTENSION_SHADOWED_BY_MEMBER","NOTHING_TO_INLINE")
public actual inline fun StringBuilder.clear(): StringBuilder = this.clear()

/**
 * Sets the character at the specified [index] to the specified [value].
 *
 * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER","NOTHING_TO_INLINE")
public actual inline fun StringBuilder.set(index: Int, value: Char) = this.set(index, value)

/**
 * Replaces characters in the specified range of this string builder with characters in the specified string [value] and returns this instance.
 * @param startIndex the beginning (inclusive) of the range to replace.
 * @param endIndex the end (exclusive) of the range to replace.
 * @param value the string to replace with.
 *
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER","NOTHING_TO_INLINE")
public actual inline fun StringBuilder.setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder =
    this.setRange(startIndex, endIndex, value)

/**
 * Removes the character at the specified [index] from this string builder and returns this instance.
 * If the `Char` at the specified [index] is part of a supplementary code point, this method does not
 * remove the entire supplementary character.
 *
 * @param index the index of `Char` to remove.
 *
 * @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER","NOTHING_TO_INLINE")
public actual inline fun StringBuilder.deleteAt(index: Int): StringBuilder = this.deleteAt(index)

/**
 * Removes characters in the specified range from this string builder and returns this instance.
 *
 * @param startIndex the beginning (inclusive) of the range to remove.
 * @param endIndex the end (exclusive) of the range to remove.
 *
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@Suppress("EXTENSION_SHADOWED_BY_MEMBER","NOTHING_TO_INLINE")
public actual inline fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder =
    this.deleteRange(startIndex, endIndex)
Open Source Used In DNAC Platform Arica 9837

string builder indices or when `startIndex > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],

* or when that index is out of the [destination] array indices range.

* @SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE", "ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")

public actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) = this.toCharArray(destination, destinationOffset, startIndex, endIndex)

/**
 * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.
 * @param value the array from which characters are appended.
 * @param startIndex the beginning (inclusive) of the subarray to append.
 * @param endIndex the end (exclusive) of the subarray to append.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")

public actual inline fun StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder = this.appendRange(value, startIndex, endIndex)

/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* @SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")

public actual inline fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder = this.insertRange(index, value, startIndex, endIndex)

*/

@SinceKotlin("1.4") @WasExperimental(ExperimentalStdlibApi::class) @Suppress("EXTENSION_SHA DOWED_BY_MEMBER", "NOTHING_TO_INLINE")
StringBuilder.insertRange(index: Int, value: CharSequence): StringBuilder =
this.insertRange(index, value)

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.text

public actual fun Char.isWhitespace(): Boolean = +this.toString().matches("[\s\xA0]") +/

@kotlin.internal.InlineOnly
public actual inline fun Char.toLowerCase(): Char =
js("String.fromCharCode")().toLowerCase().charAt(0).unsafeCast<Int>().toChar()

@kotlin.internal.InlineOnly
public actual inline fun Char.toUpperCase(): Char =
js("String.fromCharCode")().toUpperCase().charAt(0).unsafeCast<Int>().toChar()

/**
 * Returns `true` if this character is a Unicode high-surrogate code unit (also known as leading-surrogate code unit).
 */
public actual fun Char.isHighSurrogate(): Boolean = +this in Char.MIN_HIGH_SURROGATE..Char.MAX_HIGH_SURROGATE

/**
 * Returns `true` if this character is a Unicode low-surrogate code unit (also known as trailing-surrogate code unit).
 */
public actual fun Char.isLowSurrogate(): Boolean = +this in Char.MIN_LOW_SURROGATE..Char.MAX_LOW_SURROGATE

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByte(): Byte = +toByteOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toByte(radix: Int): Byte = +toByteOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an [Int] number and returns the result.
 * @throws NumberFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toInt(): Int = +toIntOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Long] number and returns the result.
 * @throws NumberFormatException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toLong(): Long = +toLongOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Double] number and returns the result.
 * @throws NumberFormatException when the string is not a valid representation of a number.
 */
public actual fun String.toDouble(): Double = +toDoubleOrNull() ?: numberFormatError(this)
String.toDouble(): Double = (+this.asDynamic()).unsafeCast<Double>().also {
    if (it.isNaN() && !this.isNaN())
        numberFormatError(this)
    || it == 0.0 && this.isBlank())
    numberFormatError(this)

/**
 * Parses the string as a [Double] number and returns the result
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toDouble(): Double =
(+(this.asDynamic())).unsafeCast<Double>().also {
    if (it.isNaN() && !this.isNaN() || it == 0.0 &&
    this.isBlank())
    numberFormatError(this)
}

/**
 * Parses the string as a [Float] number and returns the result.
 * @throws IllegalFormatException when the radix [radix] is not a valid radix for number to string conversion.
 */
public actual fun String.toFloat(): Float = toDouble().unsafeCast<Float>()

/**
 * Parses the string as a [Float] number and returns the result or `null`.
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
public actual fun String.toFloatOrNull(): Float? = toDoubleOrNull().unsafeCast<Float?>()

/**
 * Parses the string as a [Double] number and returns the result or `null`.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toDoubleOrNull(): Double? =
(+(this.asDynamic())).unsafeCast<Double>().takeIf {
    !(it.isNaN() && !this.isNaN() || it == 0.0 &&
    this.isBlank())
}

/**
 * Parses the string as a [Float] number and returns the result or `null`.
 * @throws IllegalArgumentException when the radix [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.2")
@kotlin.internal.InlineOnly
public actual inline fun String.toFloatOrNull(): Float? = toDoubleOrNull().unsafeCast<Float?>()

/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.2")
public actual fun Byte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Short] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.2")
public actual inline fun Short.toString(radix: Int): String = this.toInt().toString(checkRadix(radix))

private fun String.isNaN(): Boolean = when (this.toLowerCase()) {
    "nan", "+nan", "-nan" -> true
    else -> false
}

/**
 * Checks whether the given [radix] is valid radix for string to number and number to string conversion.
 */
@PublishedApi
internal actual fun checkRadix(radix: Int): Int {
    if (radix !in 2..36) {
        throw IllegalArgumentException("radix $radix was not in valid range 2..36")
    }
    return radix
}

internal actual fun digitOf(char: Char, radix: Int): Int = when {
    char >= '0' && char <= '9' -> char - '0'
    char >= 'A' && char <= 'Z' -> char - 'A' + 10
    char >= 'a' && char <= 'z' -> char - 'a' + 10
    else -> -1
}.let { if (it >= radix) -1 else it }

package kotlin.text

import kotlin.js.RegExp

/**
 * Provides enumeration values to use to set regular expression options.
 */
public actual enum class RegexOption(val value: String) {
    /** Enables case-insensitive matching. */
    IGNORE_CASE("i"),
    /** Enables multiline mode.
     *
     * In multiline mode the expressions `^` and `$` match just after or just before,
     * respectively, a line terminator or the end of the input sequence.
     */
    MULTILINE("m")
}

/**
 * Represents the results from a single capturing group within a [MatchResult] of [Regex].
 */
public actual data class MatchGroup(actual val value: String)

/**
 * Represents a compiled regular expression.
 * Provides functions to match strings in text with a pattern, replace the found occurrences and
 * split text around matches.
 *
 * For pattern syntax reference see [MDN RegExp](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp#Special_characters_meaning_in_regular_expressions)
 * and
 * [http://www.w3schools.com/jsref/jsref_obj_regexp.asp](https://www.w3schools.com/jsref/jsref_obj_regexp.asp).
 */
public actual constructor(pattern: String, options: Set<RegexOption>) : this(pattern, emptySet())
public actual constructor(pattern: String, option: RegexOption) : this(pattern, setOf(option))

/**
 * Creates a regular expression from the specified [pattern] string and the specified set of
 * options.
 */
public actual fun Regex(pattern: String): Regex = Regex(pattern, emptySet())

/**
 * Creates a regular expression from the specified [pattern] string and the default options.
 */
public actual fun Regex(pattern: String): Regex = Regex(pattern, emptySet())

/**
 * Creates a regular expression from the specified [pattern] string and the default options.
 */
public actual fun Regex(pattern: String): Regex = Regex(pattern, emptySet())

/**
 * Indicates whether the regular expression matches the entire [input].
 */
public actual infix fun matches(input: CharSequence): Boolean {
val match = nativePattern.exec(input.toString())
return match != null && match.index == 0 &&
nativePattern.lastIndex == input.length
/** Indicates whether the regular expression can find at least one match in the specified [input]. */
public actual fun containsMatchIn(input: CharSequence): Boolean {
    nativePattern.reset()
    return nativePattern.test(input.toString())
/** Returns the first match of a regular expression in the [input], beginning at the specified [startIndex]. */
    * @param startIndex An index to start search with, by default 0. Must be not less than zero and not greater than `input.length()` * @return An instance of [MatchResult] if match was found or `null` otherwise.
    * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of the [input] char sequence.
    */
    public actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? {
        if (startIndex < 0 || startIndex > input.length) {
            throw IndexOutOfBoundsException("Start index out of bounds: $startIndex, input length: ${input.length}")
        }
        return nativePattern.findNext(input.toString(), startIndex)
    }

    /**
     * Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the specified [startIndex].
     * @throws IndexOutOfBoundsException if [startIndex] is less than zero or greater than the length of the [input] char sequence.
     */
    public actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> {
        if (startIndex < 0 || startIndex > input.length) {
            throw IndexOutOfBoundsException("Start index out of bounds: $startIndex, input length: ${input.length}")
        }
        return generateSequence({ find(input, startIndex) }, { match -> match.next() })
    }

    /**
     * Attempts to match the entire [input] CharSequence against the pattern.
     * @return An instance of [MatchResult] if the entire input matches or `null` otherwise.
     */
    public actual fun matchEntire(input: CharSequence): MatchResult? {
        if (pattern.startsWith('^') && pattern.endsWith('$'))
            return find(input)
        else
            return Regex("^${pattern.trimStart('^').trimEnd('$')}$", options).find(input)
    }

    /**
     * Replaces all occurrences of this regular expression in the specified [input] string with specified [replacement] expression.
     */
    public actual fun replace(input: CharSequence, replacement: String): String = input.toString().nativeReplace(nativePattern, replacement)

    /**
     * Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression.
     */
    public actual fun replaceFirst(input: CharSequence, replacement: String): String {
        val nonGlobalOptions = options.map { it.value }.joinToString(separator = "")
        return input.toString().nativeReplace(nativePattern, nonGlobalOptions)
    }

    /**
     * Splits the [input] CharSequence around matches of this regular expression.
     * @param limit Non-negative value specifying the maximum number of substrings the string can be split to. * @param replace A replacement expression that can include substitutions. See [String.prototype.replace](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/replace) function docs for details.*
     */
    public actual fun split(input: CharSequence, limit: Int = 0): List<String> {
        require(limit >= 0) { "Limit must be non-negative, but was $limit" }
        return generateSequence({ find(input) }, { match -> match.next() })
            .filter { it.range.endInclusive < input.length }
            .map { input.subSequence(it.range.start, it.range.endInclusive) }
$limit" )\n    val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }\n    val result = mutableListOf<String>()\n    val lastStart = 0\n    for (match in matches) {\n        result.add(input.subSequence(lastStart, match.range.start).toString())\n        lastStart = match.range.endInclusive + 1\n    }\n    result.add(input.subSequence(lastStart, input.length).toString())\n    return result\n
/**
 * Returns the string representation of this regular expression, namely the [pattern] of this regular
 * expression.\n */\nactual override fun toString(): String = nativePattern.toString()

actual companion object {
    /**
     * Returns a regular expression that
     * matches the specified [literal] string literally.\n     * No characters of that string will have special meaning when
     * searching for an occurrence of the regular expression.
     */\n    actual fun fromLiteral(literal: String): Regex = Regex(escape(literal))

    /**
     * Returns a regular expression pattern string that matches the
     * specified [literal] string literally.\n     * No characters of that string will have special meaning when searching for
     * an occurrence of the regular expression.
     */\n    actual fun escape(literal: String): String = literal.nativeReplace(patternEscape, "\\$&")

    /**
     * Returns a literal replacement expression for the
     * specified [literal] string.
     */\n    actual fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, "$$$$")

    private val patternEscape = RegExp("""[\-\\^$*+?.()|\[\]{}]|(\\)""", "g")
    private val replacementEscape = RegExp("""\$""", "g")
}

/**
 * Creates a regular expression from the specified [pattern] string and the default options.
 */
@Deprecated("Provided for binary compatibility", level = DeprecationLevel.HIDDEN)
@JsName("Regex_61zpoe\$")
public fun Regex_1(pattern: String): Regex = Regex(pattern, emptySet())

/** Creates a regular expression from the specified
 * [pattern] string and the specified single [option].
 */
@Deprecated("Provided for binary compatibility", level = DeprecationLevel.HIDDEN)
@JsName("Regex_sb3q2\$")
public fun Regex_0(pattern: String, option: RegexOption): Regex = Regex(pattern, setOf(option))

/**
 * Returns `true` if this string starts with
 * the specified prefix.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.startsWith(prefix: String, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return nativeStartsWith(prefix, 0)
    else
        return regionMatches(0, prefix, 0, prefix.length, ignoreCase)
}

/**
 * Returns `true` if a substring of this string starting at the specified offset
 * [startIndex] starts with the specified prefix.
 */
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.startsWith(prefix: String, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase)
return nativeStartsWith(prefix, startIndex)
else
    return regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase) 

/* An @ Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS") */
public actual fun String.endsWith(suffix: String, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase)
        return nativeEndsWith(suffix)
    else
        return regionMatches(length - suffix.length, suffix, 0, suffix.length, ignoreCase) 

/* An @ Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS") */
public actual fun String.matches(regex: String): Boolean {
    return result.result !=null
}

public fun String.matches(regex: String): Boolean {
    val result = this.match(regex)
    return result != null && result.size != 0
}

public actual fun CharSequence.isBlank(): Boolean = length == 0 || (if (this is String) this else this.toString()).matches("^\s\xA0+$")

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String?.equals(other: String?, ignoreCase: Boolean = false): Boolean =
    if (this == null)
        other == null
    else if (!ignoreCase)
        this == other
    else
        other != null && other.toLowerCase() == this.toLowerCase()

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharSequence.regionMatches(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean = false): Boolean =
    regionMatchesImpl(thisOffset, other, otherOffset, length, ignoreCase)

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.capitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toUpperCase() + substring(1) else this
}

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.decapitalize(): String {
    return if (isNotEmpty()) substring(0, 1).toLowerCase() + substring(1) else this
}

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharSequence.repeat(n: Int): String {
    require(n >= 0) { "Count 'n' must be non-negative, but was $n." }
    return when (n) {
        0 -> ""
        1 -> this.toString()
        else -> {
            var result = ""
            if (!isEmpty()) {
                var s = this.toString()
                var count = n
                while (true) {
                    if ((count and 1) == 1) {
                        result += s
                    }
                    count = count ushr 1
                    if (count == 0) {
                        break
                    }
                    s += s
                }
            }
            return result
        }
    }
}

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) "gi" else "g"),
        Regex.escapeReplacement(newValue))

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "i" else ""),
        newChar.toString())

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.replaceFirst(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) "i" else ""),
        Regex.escapeReplacement(newValue))

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.replaceFirst(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "i" else ""),
        newChar.toString())

Open Source Used In DNAC Platform Arica 9842
code point corresponding to UTF-16 surrogate pair, \n * where the first of the pair is the [high] and the second is in the [string]. \n * Returns zero if the pair is malformed and [throwOnMalformed] is false. \n * @throws CharacterCodingException if the pair is malformed and [throwOnMalformed] is true. \n */
private fun codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
   val low = string[index].toInt() \n   if (low !in 0xDC00..0xDFFF) {
      return malformed(0, index, throwOnMalformed) \n   } \n   \n   return 0x10000 + ((high and 0x3FF) shl 10) or (low and 0x3FF) \n}

/**
 * Returns code point corresponding to UTF-8 sequence of two bytes, \n * where the first byte of the sequence is the [byte1] and the second byte is in the [bytes] array at the [index]. \n * Returns zero if the sequence is malformed and [throwOnMalformed] is false. \n * @throws CharacterCodingException if the sequence of two bytes is malformed and [throwOnMalformed] is true. \n */
private fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
   if (byte1 and 0x1E == 0 || index >= endIndex) {
      return malformed(0, index, throwOnMalformed) \n   } \n   val byte2 = bytes[index].toInt() \n   if (byte2 and 0xC0 != 0x80) {
      return malformed(0, index, throwOnMalformed) \n   } \n   return (byte1 shl 6) xor byte2 xor 0xF80 \n}

/**
 * Returns code point corresponding to UTF-8 sequence of three bytes, \n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index]. \n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence \n * if the sequence is malformed and [throwOnMalformed] is false. \n * @throws CharacterCodingException if the sequence of three bytes is malformed and [throwOnMalformed] is true. \n */
private fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
   if (index >= endIndex) {
      return malformed(0, index, throwOnMalformed) \n   } \n   val byte2 = bytes[index].toInt() \n   if (byte1 and 0xF == 0) {
      if (byte2 and 0xF0 <= 0x80) {
         // Non-shortest form \n         return malformed(0, index, throwOnMalformed) \n      } \n      else if (byte1 and 0xF == 0x1E || index + 1 == endIndex) {
         // Surrogate code point \n         return malformed(0, index, throwOnMalformed) \n      } \n      else if (byte2 and 0xF0 != 0x80) {
         return malformed(1, index, throwOnMalformed) \n      } \n      else if (index + 1 == endIndex) {
         return malformed(1, index, throwOnMalformed) \n      } \n      if (index + 2 == endIndex) {
         return malformed(2, index, throwOnMalformed) \n      } \n      val byte3 = bytes[index + 1].toInt() \n      if (byte3 and 0xC0 != 0x80) {
         return malformed(2, index, throwOnMalformed) \n      } \n      if (index + 3 == endIndex) {
         return malformed(3, index, throwOnMalformed) \n      } \n      val byte4 = bytes[index + 2].toInt() \n      if (byte4 and 0xC0 != 0x80) {
         return malformed(3, index, throwOnMalformed) \n      } \n      return (byte1 shl 12) xor (byte2 shl 10) xor (byte3 shl 6) xor byte4 xor 0xF800 \n}

/**
 * Returns code point corresponding to UTF-8 sequence of four bytes, \n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index]. \n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence \n * if the sequence is malformed and [throwOnMalformed] is false. \n * @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true. \n */
   if (index >= endIndex) {
      return malformed(0, index, throwOnMalformed) \n   } \n   val byte2 = bytes[index].toInt() \n   if (byte1 and 0xF == 0) {
      if (byte2 and 0xF0 <= 0x80) {
         // Non-shortest form \n         return malformed(0, index, throwOnMalformed) \n      } \n      else if (byte1 and 0xF == 0x1E || index + 1 == endIndex) {
         // Out of Unicode code points domain (larger than U+10FFFF) \n         return malformed(0, index, throwOnMalformed) \n      } \n      else if (byte2 and 0xF0 > 0x84) {
         return malformed(0, index, throwOnMalformed) \n      } \n      else if (byte2 and 0xC0 != 0x80) {
         return malformed(1, index, throwOnMalformed) \n      } \n      if (index + 1 == endIndex) {
         return malformed(1, index, throwOnMalformed) \n      } \n      val byte3 = bytes[index + 1].toInt() \n      if (byte3 and 0xC0 != 0x80) {
         return malformed(1, index, throwOnMalformed) \n      } \n      if (index + 2 == endIndex) {
         return malformed(2, index, throwOnMalformed) \n      } \n      val byte4 = bytes[index + 2].toInt() \n      if (byte4 and 0xC0 != 0x80) {
         return malformed(2, index, throwOnMalformed) \n      } \n      return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0xF8000 \n}}

Open Source Used In DNAC Platform Arica 9843
Open Source Used In DNAC Platform Arica

*0x10000..0x10FFFF* are represented by a pair of surrogate `Char`s and are encoded in four bytes.

```kotlin
private const val MAX_BYTES_PER_CHAR = 3

/**
 * The byte sequence a malformed UTF-16 char sequence is
 * replaced by.
 */
private val REPLACEMENT_BYTE_SEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(),
  0xBF.toByte(), 0xBD.toByte())

/**
 * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].
 */
internal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {
  val byteIndex = startIndex
  val charIndex = startIndex
  var byteIndex = 0
  var charIndex = startIndexOf
  while (charIndex < endIndex) {
    val code = string[charIndex++].toInt()
    when {
      code < 0x80 ->
        byteIndex++
      code < 0x800 -> {
        byteIndex++
        byteIndex++ = ((code shr 6) or 0x80).toByte()
      }
      code < 0xD800 || code >= 0xE000 -> {
        byteIndex++
        byteIndex++ = (((code shr 12) and 0x3F) or 0x80).toByte()
        byteIndex++ = ((code and 0x3F) or 0x80).toByte()
      }
      else -> { // Surrogate char value
        val codePoint = codePointFromSurrogate(string, code, charIndex, endIndexOf, throwOnMalformed)
        if (codePoint <= 0) {
          byteIndex++ = REPLACEMENT_BYTE_SEQUENCE[0]
          byteIndex++ = REPLACEMENT_BYTE_SEQUENCE[1]
          byteIndex++ = REPLACEMENT_BYTE_SEQUENCE[2]
        } else {
          byteIndex++ = ((codePoint shr 18) or 0xF0).toByte()
          byteIndex++ = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
          byteIndex++ = (((codePoint shr 6) and 0x3F) or 0x80).toByte()
          byteIndex++ = (codePoint and 0x3F) or 0x80
        }
      }
    }
  }
  return byteIndex == byteIndex
}

/**
 * The character a malformed UTF-8 byte sequence is replaced by.
 */
private const val REPLACEMENT_CHAR = '\uFFFD'

/**
 * Decodes the UTF-8 [bytes] array and returns the resulting [String].
 */
internal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {
  val byteIndex = startIndex
  val charIndex = startIndex
  while (byteIndex < endIndex) {
    val byte = bytes[byteIndex++]
    when {
      byte >= 0 ->
        stringBuilder.append(byte.toChar())
      byte shr 5 == -2 -> {
        val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)
        if (code <= 0) {
          stringBuilder.append(REPLACEMENT_CHAR)
          byteIndex += -code
        } else {
          stringBuilder.append(code.toChar())
          byteIndex += 1
        }
      }
      byte shr 4 == -2 -> {
        val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)
        if (code <= 0) {
          stringBuilder.append(REPLACEMENT_CHAR)
          byteIndex += -code
        } else {
          stringBuilder.append(code.toChar())
          byteIndex += 2
        }
      }
      byte shr 3 == -2 -> {
        val code = codePointFrom4(bytes, byte, byteIndex, endIndex, throwOnMalformed)
        if (code <= 0) {
          stringBuilder.append(REPLACEMENT_CHAR)
          byteIndex += -code
        } else {
          stringBuilder.append(code.toChar())
          byteIndex += 3
        }
      }
      else -> {
        val high = (code - 0x10000) shr 10 or 0xD800
        val low = (code and 0x3FF) or 0xDC00
        stringBuilder.append(high.toChar())
        charIndex++
        byteIndex++ = (codePoint shr 18) or 0xF0
        byteIndex++ = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
        byteIndex++ = (((codePoint shr 6) and 0x3F) or 0x80).toByte()
        byteIndex++ = (codePoint and 0x3F) or 0x80
      }
    }
  }
  return stringBuilder.toString()
}
Open Source Used In DNAC Platform Arica

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Returns the detailed description of this throwable with its stack trace.
 * The detailed description includes:
 * - the short description (see Throwable.toString()) of this throwable;
 * - the complete stack trace;
 * - detailed descriptions of the exceptions that were suppressed;
 * - the detailed description of each throwable in the Throwable.caused chain.
 */

fun Throwable.stackTraceToString(): String = ExceptionTraceBuilder().buildFor(this)

fun Throwable.printStackTrace() {
    console.error(this.stackTraceToString())
}

fun Throwable.addSuppressed(exception: Throwable) {
    if (this !== exception) {
        val suppressed = this.asDynamic()._suppressed.unsafeCast<MutableList<Throwable>?>()
        if (suppressed == null) {
            this.asDynamic()._suppressed = mutableListOf(exception)
        } else {
            suppressed.add(exception)
        }
    }
}

fun Throwable.suppressedExceptions: List<Throwable>
    get() {
        return this.asDynamic()._suppressed?.unsafeCast<List<Throwable>>() ?: emptyList()
    }

private class ExceptionTraceBuilder {
    private val target = StringBuilder()
    private val visited = arrayOf<Throwable>()
    private var topStack: String = ""
    private var topStackStart: Int = 0

    fun buildFor(exception: Throwable): String {
        exception.dumpFullTrace("", "")
        return target.toString()
    }

    private fun hasSeen(exception: Throwable): Boolean = visited.any { it === exception }

    private fun Throwable.dumpSelfTrace(indent: String, qualifier: String): Boolean {
        target.append(indent).append(qualifier)
        val shortInfo = this.toString()
        if (hasSeen(this)) {
            target.append("[CIRCULAR REFERENCE, SEE ABOVE:"]
            if (shortInfo.isEmpty()) {
                return false
            } else {
                target.append(shortInfo).append("\n")
            }
        }
        visited.asDynamic().push(this)

        var stack = this.asDynamic().stack as String?
        if (stack != null) {
            val stackStart = stack.indexOf(shortInfo).let { if (it < 0) 0 else it + shortInfo.length }
            if (stackStart == 0) {
                target.append(shortInfo).append("\n")
            } else {
                stack = dropCommonFrames(stack, stackStart)
            }
        }

        if (indent.isNotEmpty()) {
            // indent stack, but avoid indenting exception message lines
            val messageLines = if (stackStart == 0) 0 else 1 + shortInfo.count { it == '\n' }
            stack.lineSequence().forEachIndexed { index: Int, line: String ->
                if (index >= messageLines) {
                    target.append(indent).append(line).append("\n")
                }
            }
        } else {
            target.append(stack).append("\n")
        }

        val suppressed = suppressedExceptions
        if (suppressed.isNotEmpty()) {
            target.append("Suppressed:")
            for (s in suppressed) {
                s.dumpFullTrace(indent,"")
            }
        }
        return true
    }

    private fun dropCommonFrames(stack: String, stackStart: Int): String {
        var commonFrames: Int = 0
        var lastBreak: Int = 0
        var preLastBreak: Int = 0
        for (pos in 0 until minOf(topStack.length - topStackStart, stack.length - stackStart)) {
            val c = stack[stackStart + pos]
            if (c != topStack[topStackStart + pos]) break
            if (c == '\n') {
                commonFrames += 1
                preLastBreak = lastBreak
                lastBreak = pos
            } else {
                target.append(stack).append("...")
                break
            }
        }

        if (commonFrames <= 1) return stack
        return suppressedIndent = indent + "\n"
        while (preLastBreak > 0 && stack[stackStart + preLastBreak - 1] == ' ')
            preLastBreak -= 1
        return suppressed = suppressedExceptions
    }

    private fun Throwable.dumpFullTrace(indent: String, qualifier: String) {
        this.dumpSelfTrace(indent,"")
        val shortInfo = this.toString()
        if (hasSeen(this)) {
            target.append("[CIRCULAR REFERENCE, SEE ABOVE:"]
            if (shortInfo.isEmpty()) {
                return false
            } else {
                target.append(shortInfo).append("\n")
            }
        }
        visited.asDynamic().push(this)

        var stack = this.asDynamic().stack as String?
        if (stack != null) {
            val stackStart = stack.indexOf(shortInfo).let { if (it < 0) 0 else it + shortInfo.length }
            if (stackStart == 0) {
                target.append(shortInfo).append("\n")
            } else {
                stack = dropCommonFrames(stack, stackStart)
            }
        }

        if (indent.isNotEmpty()) {
            // indent stack, but avoid indenting exception message lines
            val messageLines = if (stackStart == 0) 0 else 1 + shortInfo.count { it == '\n' }
            stack.lineSequence().forEachIndexed { index: Int, line: String ->
                if (index >= messageLines) {
                    target.append(indent).append(line).append("\n")
                }
            }
        } else {
            target.append(stack).append("\n")
        }

        val suppressed = suppressedExceptions
        if (suppressed.isNotEmpty()) {
            target.append("Suppressed:")
            for (s in suppressed) {
                s.dumpFullTrace(indent,"")
            }
        }
        return true
    }

    private fun dropCommonFrames(stack: String, stackStart: Int): String {
        var commonFrames: Int = 0
        var lastBreak: Int = 0
        var preLastBreak: Int = 0
        for (pos in 0 until minOf(topStack.length - topStackStart, stack.length - stackStart)) {
            val c = stack[stackStart + pos]
            if (c != topStack[topStackStart + pos]) break
            if (c == '\n') {
                commonFrames += 1
                preLastBreak = lastBreak
                lastBreak = pos
            } else {
                target.append(stack).append("...")
                break
            }
        }

        if (commonFrames <= 1) return stack
        return suppressedIndent = indent + "\n"
        while (preLastBreak > 0 && stack[stackStart + preLastBreak - 1] == ' ')
            preLastBreak -= 1
        return suppressed = suppressedExceptions
    }
}
source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
package kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
public actual enum class DurationUnit(internal val scale: Double) {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS(1e0),
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS(1e3),
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILLISECONDS(1e6),
    /**
     * Time unit representing one second.
     */
    SECONDS(1e9),
    /**
     * Time unit representing one minute.
     */
    MINUTES(60e9),
    /**
     * Time unit representing one hour.
     */
    HOURS(3600e9),
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS(86400e9)
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {
    val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)
    return when {
        sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)
        sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale)
        else -> value
    }
}

import org.w3c.performance.

@SinceKotlin("1.3")
@ExperimentalTime
internal actual object MonotonicTimeSource : TimeSource {
    private val actualSource: TimeSource = when {
        isNode -> HrTimeSource(js("process").unsafeCast<Process>())
        else -> js("self").unsafeCast<GlobalPerformance?>()?.performance?.let(::PerformanceTimeSource)
    }

    override fun markNow(): TimeMark = actualSource.markNow()
}

class HrTimeSource(val process: Process) : TimeSource {
    override fun markNow(): TimeMark = object : TimeMark() {
        val startedAt = process.hrtime()
        override fun elapsedNow(): Duration = process.hrtime(startedAt).let { (seconds, nanos) -> seconds.seconds + nanos.nanoseconds }
    }
}

class PerformanceTimeSource(val performance: Performance) : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = performance.now()
    override fun toString(): String = "TimeSource(self.performance.now())"
}

object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = kotlin.js.Date.now()
    override fun toString(): String = "TimeSource(Date.now())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun formatToExactDecimals(value: Double, decimals: Int): String {
    val rounded = if (decimals == 0) {
        value
    } else {
        val pow = 10.0.pow(decimals)
        kotlin.js.Math.round(abs(value) * pow) / pow * sign(value)
    }
    return rounded.asDynamic().toFixed(decimals).unsafeCast<String>()
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun formatUpToDecimals(value: Double, decimals: Int): String {
    return value.asDynamic().toLocaleString("en-us", json("maximumFractionDigits" to decimals)).unsafeCast<String>()
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun formatScientific(value: Double): String {
    return value.asDynamic().toExponential(2).unsafeCast<String>()
}
```

Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors

We use this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
the specified [init] function.

```kotlin
public fun Document.createElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) } // returns a newly created element with the specified [name] to this element.

    * The element is initialized with the specified [init] function.

    // * An * Element.appendElement(name: String, init: Element.() -> Unit): Element {
    contract { callsInPlace(init, InvocationKind.EXACTLY_ONCE) } // returns the ownerDocument!! of the newly created element.

    * * Element.appendChild(it) } } 
```

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

import kotlin.contracts.*

/**
 * A mutable sequence of characters.
 * 
 * String builder can be used to efficiently perform multiple string manipulation operations.
 */
expect class StringBuilder : Appendable, CharSequence {
    /** Constructs an empty string builder. */
    constructor()

    /** Constructs an empty string builder with the specified initial [capacity]. */
    constructor(capacity: Int)

    /** Constructs a string builder that contains the same characters as the specified [content] string. */
    constructor(content: String)

    /** Constructs a string builder that contains the same characters as the specified [content] sequence. */
    constructor(content: CharSequence)

    /** Reverses the contents of this string builder and returns this instance. */
    fun reverse(): StringBuilder

    private fun reverse(): StringBuilder

    override val length: Int

    override operator fun get(index: Int): Char

    override fun subSequence(startIndex: Int, endIndex: Int): CharSequence

    override fun append(value: Char): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder

    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder
```

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

import kotlin.contracts.*

/**
 * A mutable sequence of characters.
 * 
 * String builder can be used to efficiently perform multiple string manipulation operations.
 */
expect class StringBuilder : Appendable, CharSequence {
    /** Constructs an empty string builder. */
    constructor()

    /** Constructs an empty string builder with the specified initial [capacity]. */
    constructor(capacity: Int)

    /** Constructs a string builder that contains the same characters as the specified [content] string. */
    constructor(content: String)

    /** Constructs a string builder that contains the same characters as the specified [content] sequence. */
    constructor(content: CharSequence)

    /** Reverses the contents of this string builder and returns this instance. */
    fun reverse(): StringBuilder

    private fun reverse(): StringBuilder

    override val length: Int

    override operator fun get(index: Int): Char

    override fun subSequence(startIndex: Int, endIndex: Int): CharSequence

    override fun append(value: Char): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder

    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder
```

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

import kotlin.contracts.*

/**
 * A mutable sequence of characters.
 * 
 * String builder can be used to efficiently perform multiple string manipulation operations.
 */
expect class StringBuilder : Appendable, CharSequence {
    /** Constructs an empty string builder. */
    constructor()

    /** Constructs an empty string builder with the specified initial [capacity]. */
    constructor(capacity: Int)

    /** Constructs a string builder that contains the same characters as the specified [content] string. */
    constructor(content: String)

    /** Constructs a string builder that contains the same characters as the specified [content] sequence. */
    constructor(content: CharSequence)

    /** Reverses the contents of this string builder and returns this instance. */
    fun reverse(): StringBuilder

    private fun reverse(): StringBuilder

    override val length: Int

    override operator fun get(index: Int): Char

    override fun subSequence(startIndex: Int, endIndex: Int): CharSequence

    override fun append(value: Char): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?): StringBuilder

    override fun append(value: String): StringBuilder
    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder

    override fun append(value: CharSequence?, start: Int, end: Int): StringBuilder
```
""\uDC00\uDC00\uDC00\uDC00" produces "\uDC00\uDC00\uDC00\uDC00" which is a valid surrogate pair.

fun reverse():
    StringBuilder

     * Appends the string representation of the specified object [value] to this string builder and returns this instance.
     * The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method, and then that string was appended to this string builder.
     * Appends characters in the specified character array [value] to this string builder and returns this instance.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun append(value: Any?): StringBuilder

        /\n        * Appends the specified string [value] to this string builder and returns this instance.
        * If [value] is `null`, then the four characters `\null\null\null\null` are appended.

    @SinceKotlin("1.3")
    fun append(value: CharArray): StringBuilder

        /\n        * Returns the current capacity of this string builder.
        * The capacity is the maximum length this string builder can have before an allocation occurs.
        * If the current capacity is less than the [minimumCapacity], a new backing storage is allocated with greater capacity.
        * Otherwise, this method takes no action and simply returns.

    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun append(value: Char): StringBuilder

        /\n        * Appends the specified character [value] to this string builder and returns this instance.

    @SinceKotlin("1.4")
    fun insert(index: Int, value: Boolean): StringBuilder

        /\n        * Inserts the string representation of the specified boolean [value] to this string builder and returns this instance.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun append(value: Boolean): StringBuilder

        /\n        * Appends the string representation of the specified boolean [value] to this string builder and returns this instance.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun append(value: String?): StringBuilder

        /\n        * Appends the specified string [value] to this string builder and returns this instance.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun ensureCapacity(minimumCapacity: Int)

        /\n        * Ensures that the capacity of this string builder is at least equal to the specified [minimumCapacity].
        * If the current capacity is less than the [minimumCapacity], a new backing storage is allocated with greater capacity.

    @ExperimentalStdlibApi
    @Deprecated("Obtaining StringBuilder capacity is not supported in JS and common code.", level = DeprecationLevel.ERROR)
    fun capacity(): Int

        /\n        * Returns the current capacity of this string builder.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun lastIndexOf(string: String, startIndex: Int): Int

        /\n        * Returns the last occurrence of the specified string in this string builder.
        * The last occurrence of empty string `""` is considered to be at the index equal to `this.length`.
        * Returns `-1` if the specified string does not occur in this string builder.

    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun lastIndexOf(string: String): Int

        /\n        * Returns the last occurrence of the specified string in this string builder.
        * If the specified string does not occur in this string builder, this method takes no action and simply returns.

    @SinceKotlin("1.3")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun indexOf(string: String, startIndex: Int): Int

        /\n        * Returns the index within this string builder of the last occurrence of the specified [string].
        * Returns `-1` if the specified [string] does not occur in this string builder.

    @SinceKotlin("1.4")
    @WasExperimental(ExperimentalStdlibApi::class)
    fun indexOf(string: String): Int

        /\n        * Returns the index within this string builder of the first occurrence of the specified [string].
        * Returns `-1` if the specified [string] does not occur in this string builder.

    @SinceKotlin("1.3")
    fun reverse(): StringBuilder

        /\n        * Reverses the characters in this string builder.

    @SinceKotlin("1.4")
    fun capacity(): Int

        /\n        * Returns the current capacity of this string builder.
character at the specified [index] from this string builder and returns this instance.

* @param index the index of ‘Char’ to remove.
* @throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.

* @param startIndex the beginning (inclusive) of the range to remove.
* @param endIndex the end (exclusive) of the range to remove.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* Append characters from this string builder into the [destination] character array.
* @param destination the array to copy to.
* @param destinationOffset the position in the array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @param index the position in this string builder to insert at.
* @param startIndex the beginning (inclusive) of the subarray to insert.
* @param endIndex the end (exclusive) of the subarray to insert.
* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subarray of the specified character array [value] into this string builder and returns this instance.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

* Insert characters in a subsequence of the specified character sequence [value] into this string builder and returns this instance.
sequence, starting at [index].

* @param index the position in this string builder to insert at.

* @param o the character sequence to insert.

* @throws IndexOutOfBoundsException when [index] is out of range of the [value] character sequence indices or when [index] is less than zero or greater than the length of this string builder.

* @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when [startIndex] > [endIndex].

* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when [startIndex] > [endIndex].

* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.

* @param endIndex the end (exclusive) of the subsequence to insert.

* @param value the character sequence from which a subsequence is inserted.

* @param startIndex the beginning (inclusive) of the subsequence to insert.
license/LICENSE.txt file.
 *
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 *
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

/**
 * Exposes the JavaScript
 * WebGLBuffer(https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer) to Kotlin
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

/**
 * Exposes the JavaScript
 * WebGLFrameBuffer(https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer) to Kotlin
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

/**
 * Exposes the JavaScript
 * WebGLProgram(https://developer.mozilla.org/en/docs/Web/API/WebGLProgram) to Kotlin
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

/**
 * Exposes the JavaScript
 * WebGLRenderer(https://developer.mozilla.org/en/docs/Web/API/WebGLRenderer) to Kotlin
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

/**
 * Exposes the JavaScript
 * WebGLShader(https://developer.mozilla.org/en/docs/Web/API/WebGLShader) to Kotlin
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

Open Source Used In DNAC Platform Arica
fun getProgramParameter(program: WebGLProgram?, pname: Int): Any?
fun getProgramInfoLog(program: WebGLProgram?): String?
fun getRenderbufferParameter(target: Int, pname: Int): Any?
fun getShaderParameter(shader: WebGLShader?, pname: Int): Any?
fun getShaderPrecisionFormat(shadertype: Int, precisiontype: Int): WebGLShaderPrecisionFormat?
fun getShaderInfoLog(shader: WebGLShader?): String?
fun getShaderSource(shader: WebGLShader?): String?
fun getProgramInfoLog(shader: WebGLShader?): String?
fun getUniform(program: WebGLProgram?, location: WebGLUniformLocation?, pname: Int): Any?
fun getUniformLocation(program: WebGLProgram?, name: String): WebGLUniformLocation?
fun getVertexAttrib(index: Int, pname: Int): Any?
fun getVertexAttribOffset(index: Int, pname: Int): Int
fun hint(target: Int, mode: Int)
fun isBuffer(buffer: WebGLBuffer?): Boolean
fun isEnabled(cap: Int): Boolean
fun isFramebuffer(framebuffer: WebGLFramebuffer?): Boolean
fun isProgram(program: WebGLProgram?): Boolean
fun isRenderbuffer(renderbuffer: WebGLRenderbuffer?): Boolean
fun isTexture(texture: WebGLTexture?): Boolean
fun lineWidth(width: Float)
fun linkProgram(program: WebGLProgram?)
fun pixelStorei(pname: Int, param: Int)
fun polygonOffset(factor: Float, units: Float)
fun readPixels(x: Int, y: Int, width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?)
fun renderbufferStorage(target: Int, internalformat: Int, width: Int, height: Int)
fun sampleCoverage(value: Float, invert: Boolean)
fun scissor(x: Int, y: Int, width: Int, height: Int)
fun shaderSource(shader: WebGLShader?, source: String)
fun stencilFunc(func: Int, ref: Int, mask: Int)
fun stencilFuncSeparate(face: Int, func: Int, ref: Int, mask: Int)
fun stencilMask(mask: Int)
fun stencilMaskSeparate(face: Int, mask: Int)
fun stencilOp(fail: Int, zfail: Int, zpass: Int)
fun texImage2D(target: Int, level: Int, internalformat: Int, format: Int, type: Int, source: TexImageSource?)
fun texParameterf(target: Int, pname: Int, param: Float)
fun texParameteri(target: Int, pname: Int, param: Int)
fun uniform1f(location: WebGLUniformLocation?, x: Float)
fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform1fv(location: WebGLUniformLocation?, v: Array<Float>)
fun uniform1i(location: WebGLUniformLocation?, x: Int)
fun uniform1iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform1iv(location: WebGLUniformLocation?, v: Array<Int>)
fun uniform2f(location: WebGLUniformLocation?, x: Float, y: Float)
fun uniform2fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform2fv(location: WebGLUniformLocation?, v: Array<Float>)
fun uniform2i(location: WebGLUniformLocation?, x: Int, y: Int)
fun uniform2iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform2iv(location: WebGLUniformLocation?, v: Array<Int>)
fun uniform3f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float)
fun uniform3fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform3fv(location: WebGLUniformLocation?, v: Array<Float>)
fun uniform3i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int)
fun uniform3iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform3iv(location: WebGLUniformLocation?, v: Array<Int>)
fun uniform4f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float, w: Float)
fun uniform4fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform4fv(location: WebGLUniformLocation?, v: Array<Float>)
fun uniform4i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int, w: Int)
fun uniform4iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform4iv(location: WebGLUniformLocation?, v: Array<Int>)
fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)
fun uniformMatrix2fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)
fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)
fun uniformMatrix3fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)
fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value: Float32Array)
fun uniformMatrix4fv(location: WebGLUniformLocation?, transpose: Boolean, value: Array<Float>)
Open Source Used In DNAC Platform Arica  9855

```
useProgram(program: WebGLProgram?)
fun validateProgram(program: WebGLProgram?)
fun vertexAttrib1f(index: Int, x: Float)
fun vertexAttrib1fv(index: Int, values: dynamic)
fun vertexAttrib2f(index: Int, x: Float, y: Float)
fun vertexAttrib2fv(index: Int, values: dynamic)
fun vertexAttrib3f(index: Int, x: Float, y: Float, z: Float)
fun vertexAttrib3fv(index: Int, values: dynamic)
fun vertexAttrib4f(index: Int, x: Float, y: Float, z: Float, w: Float)
fun vertexAttrib4fv(index: Int, values: dynamic)
fun viewport(x: Int, y: Int, width: Int, height: Int)

companion object {
    val DEPTH_BUFFER_BIT: Int
    val STENCIL_BUFFER_BIT: Int
    val COLOR_BUFFER_BIT: Int
    val POINTS: Int
    val LINES: Int
    val LINE_LOOP: Int
    val LINE_STRIP: Int
    val TRIANGLES: Int
    val TRIANGLE_STRIP: Int
    val TRIANGLE_FAN: Int
    val ZERO: Int
    val ONE: Int
    val SRC_COLOR: Int
    val SRC_ALPHA: Int
    val DST_ALPHA: Int
    val ONE_MINUS_DST_ALPHA: Int
    val DST_COLOR: Int
    val SRC_ALPHA_SATURATE: Int
    val BLEND_EQUATION: Int
    val BLEND_EQUATION_RGB: Int
    val BLEND_EQUATION_ALPHA: Int
    val FUNC_ADD: Int
    val BLEND_EQUATION: Int
    val FUNC_SUBTRACT: Int
    val FUNC_REVERSE_SUBTRACT: Int
    val BLEND_SRC_RGB: Int
    val BLEND_SRC_ALPHA: Int
    val BLEND_DST_RGB: Int
    val BLEND_DST_ALPHA: Int
    val BLEND_COLOR: Int
    val ARRAY_BUFFER: Int
    val ELEMENT_ARRAY_BUFFER: Int
    val ARRAY_BUFFER_BINDING: Int
    val ELEMENT_ARRAY_BUFFER_BINDING: Int
    val STREAM_DRAW: Int
    val STATIC_DRAW: Int
    val DYNAMIC_DRAW: Int
    val BUFFER_SIZE: Int
    val BUFFER_USAGE: Int
    val CURRENT_VERTEX_ATTRIB: Int
    val FRONT: Int
    val BACK: Int
    val FRONT_AND_BACK: Int
    val CULL_FACE: Int
    val BLEND: Int
    val DITHER: Int
    val STENCIL_TEST: Int
    val DEPTH_TEST: Int
    val SCISSOR_TEST: Int
    val POLYGON_OFFSET_FILL: Int
    val SAMPLE_ALPHA_TO_COVERAGE: Int
    val SAMPLE_COVERAGE: Int
    val SAMPLE_BUFFERS: Int
    val SAMPLE_COVERAGE_VALUE: Int
    val SAMPLE_COVERAGE_INVERT: Int
    val COMPRESSED_TEXTURE_FORMATS: Int
    val DONT_CARE: Int
    val FASTEST: Int
    val NICEST: Int
    val GENERATE_MIPMAP_HINT: Int
    val BYTE: Int
    val UNSIGNED_BYTE: Int
    val SHORT: Int
    val UNSIGNED_SHORT: Int
    val INT: Int
    val UNSIGNED_INT: Int
    val POLYGON_OFFSET_UNITS: Int
    val POLYGON_OFFSET_FACTOR: Int
    val TEXTURE_BINDING_2D: Int
    val MAX_TEXTURE_SIZE: Int
    val MAX_VIEWPORT_DIMS: Int
    val SUBPIXEL_BITS: Int
    val RED_BITS: Int
    val GREEN_BITS: Int
    val BLUE_BITS: Int
    val ALPHA_BITS: Int
    val DEPTH_BITS: Int
    val STENCIL_BITS: Int
    val PACK_ALIGNMENT: Int
    val MAX_TEXTURE_SIZE: Int
    val MAX_VIEWPORT_DIMS: Int
    val SUBPIXEL_BITS: Int
    val RED_BITS: Int
    val GREEN_BITS: Int
    val BLUE_BITS: Int
    val ALPHA_BITS: Int
    val DEPTH_BITS: Int
    val STENCIL_BITS: Int
    val TEXTURE_BINDING: Int
    val UNPACK_ALIGNMENT: Int
    val MAX_TEXTURE_SIZE: Int
    val MAX_VIEWPORT_DIMS: Int
```

---

Open Source Used In DNAC Platform Arica  9855
RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_BLUE_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_BINDING: Int
val RENDERTBUFFER_BINDING: Int
val MAX_RENDERBUFFER_SIZE: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
val BROWSER_DEFAULT_WEBGL: Int
val DEPTH_BUFFER_BIT: Int
val STENCIL_BUFFER_BIT: Int
val COLOR_BUFFER_BIT: Int
val POINTS: Int
val LINES: Int
val LINE_LOOP: Int
val LINE_STRIP: Int
val TRIANGLES: Int
val TRIANGLE_STRIP: Int
val SRC_COLOR: Int
val ONE_MINUS_SRC_COLOR: Int
val SRC_ALPHA: Int
val ONE_MINUS_SRC_ALPHA: Int
val DST_ALPHA: Int
val ONE_MINUS_DST_ALPHA: Int
val DST_COLOR: Int
val ONE_MINUS_DST_COLOR: Int
val SRC_ALPHA_SATURATE: Int
val FUNC_ADD: Int
val BLEND_EQUATION: Int
val BLEND_EQUATION_RGB: Int
val BLEND_EQUATION_ALPHA: Int
val FUNC_SUBTRACT: Int
val BLEND_SRC_RGB: Int
val BLEND_SRC_ALPHA: Int
val BLEND_DST_RGB: Int
val BLEND_DST_ALPHA: Int
val CONSTANT_COLOR: Int
val ONE_MINUS_CONSTANT_COLOR: Int
val CONSTANT_ALPHA: Int
val ONE_MINUS_CONSTANT_ALPHA: Int
val BLEND_COLOR: Int
val ARRAY_BUFFER: Int
val ELEMENT_ARRAY_BUFFER: Int
val STREAM_DRAW: Int
val STATIC_DRAW: Int
val BUFFER_SIZE: Int
val CURRENT_VERTEX_ATTRIB: Int
val FRONT: Int
val BACK: Int
val CULL_FACE: Int
val BLEND: Int
val DITHER: Int
val STENCIL_TEST: Int
val DEPTH_TEST: Int
val SCISSOR_TEST: Int
val POLYGON_OFFSET_FILL: Int
val SAMPLE_ALPHA_TO_COVERAGE: Int
val NO_ERROR: Int
val INVALID_ENUM: Int
val INVALID_OPERATION: Int
val OUT_OF_MEMORY: Int
val CW: Int
val CCW: Int
val LINE_WIDTH: Int
val ALIASED_POINT_SIZE_RANGE: Int
val ALIASED_LINE_WIDTH_RANGE: Int
val CULL_FACE_MODE: Int
val FRONT_FACE: Int
val DEPTH_RANGE: Int
val DEPTH_WRITEMASK: Int
val DEPTH_CLEAR_VALUE: Int
val DEPTH_FUNC: Int
val STENCIL_CLEAR_VALUE: Int
val STENCIL_FUNC: Int
val STENCIL_PASS_DEPTH_FAIL: Int
val STENCIL_PASS_DEPTH_PASS: Int
val STENCIL_REF: Int
val STENCIL_VALUE_MASK: Int
val STENCIL_BACK_FUNC: Int
val STENCIL_BACK_FAIL: Int
val STENCIL_BACK_PASS_DEPTH_FAIL: Int
val STENCIL_BACK_PASS_DEPTH_PASS: Int
val STENCIL_BACK_REF: Int
val STENCIL_BACK_VALUE_MASK: Int
val
STENCIL_BACK_WRITEMASK: Int
        val VIEWPORT: Int
        val SCISSOR_BOX: Int
        val COLOR_CLEAR_VALUE: Int
        val COLOR_WRITEMASK: Int
        val UNPACK_ALIGNMENT: Int
        val PACK_ALIGNMENT: Int
        val MAX_TEXTURE_SIZE: Int
        val MAX_VIEWPORT_DIMS: Int
        val SUBPIXEL_BITS: Int
        val RED_BITS: Int
        val GREEN_BITS: Int
        val BLUE_BITS: Int
        val ALPHA_BITS: Int
        val DEPTH_BITS: Int
        val STENCIL_BITS: Int
        val POLYGON_OFFSET_UNITS: Int
        val POLYGON_OFFSET_FACTOR: Int
        val TEXTURE_BINDING_2D: Int
        val SAMPLE_BUFFERS: Int
        val SAMPLES: Int
        val SAMPLE_COVERAGE_VALUE: Int
        val SAMPLE_COVERAGE_INVERT: Int
        val COMPRESSED_TEXTURE_FORMATS: Int
        val DONT_CARE: Int
        val FASTEST: Int
        val NICEST: Int
        val GENERATE_MIPMAP_HINT: Int
        val BYTE: Int
        val UNSIGNED_BYTE: Int
        val SHORT: Int
        val UNSIGNED_SHORT: Int
        val INT: Int
        val UNSIGNED_INT: Int
        val FLOAT: Int
        val DEPTH_COMPONENT: Int
        val ALPHA: Int
        val RGB: Int
        val RGBA: Int
        val LUMINANCE: Int
        val LUMINANCE_ALPHA: Int
        val UNSIGNED_SHORT_4_4_4_4: Int
        val UNSIGNED_SHORT_5_5_5_1: Int
        val UNSIGNED_SHORT_5_6_5: Int
        val FRAGMENT_SHADER: Int
        val VERTEX_SHADER: Int
        val MAX_VERTEX_ATTRIBS: Int
        val MAX_VERTEX_UNIFORM_VECTORS: Int
        val MAX_VARYING_VECTORS: Int
        val MAX_COMBINED_TEXTURE_IMAGE_UNITS: Int
        val MAX_VERTEX_TEXTURE_IMAGE_UNITS: Int
        val MAX_TEXTURE_IMAGE_UNITS: Int
        val MAX_FRAGMENT_UNIFORM_VECTORS: Int
        val SHADER_TYPE: Int
        val DELETE_STATUS: Int
        val ACTIVE_UNIFORMS: Int
        val ACTIVE_ATTRIBUTES: Int
        val SHADING_LANGUAGE_VERSION: Int
        val CURRENT_PROGRAM: Int
        val NEVER: Int
        val LESS: Int
        val EQUAL: Int
        val LEQUAL: Int
        val GREATER: Int
        val NOTEQUAL: Int
        val GEQUAL: Int
        val ALWAYS: Int
        val KEEP: Int
        val REPLACE: Int
        val INCR: Int
        val DECR: Int
        val INVERT: Int
        val INCR_WRAP: Int
        val DECR_WRAP: Int
        val VENDOR: Int
        val RENDERER: Int
        val VERSION: Int
        val TEXTURE_MAG_FILTER: Int
        val TEXTURE_MIN_FILTER: Int
        val TEXTURE_WRAP_S: Int
        val TEXTURE_WRAP_T: Int
        val TEXTURE_CUBE_MAP: Int
        val TEXTURE_BINDING_CUBE_MAP: Int
        val TEXTURE_CUBE_MAP_POSITIVE_X: Int
        val TEXTURE_CUBE_MAP_NEGATIVE_X: Int
        val TEXTURE_CUBE_MAP_POSITIVE_Y: Int
        val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
        val TEXTURE_CUBE_MAP_POSITIVE_Z: Int
        val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int
        val TEXTURE_CUBE_MAP_TEXTURE_SIZE: Int
        val TEXTURE0: Int
        val TEXTURE1: Int
        val TEXTURE2: Int
        val TEXTURE3: Int
        val TEXTURE4: Int
        val TEXTURE5: Int
        val TEXTURE6: Int
        val TEXTURE7: Int
        val TEXTURE8: Int
        val TEXTURE9: Int
        val TEXTURE10: Int
        val TEXTURE11: Int
        val TEXTURE12: Int
        val TEXTURE13: Int
        val TEXTURE14: Int
        val TEXTURE15: Int
        val TEXTURE16: Int
        val TEXTURE17: Int
        val TEXTURE18: Int
        val TEXTURE19: Int
        val TEXTURE20: Int
        val TEXTURE21: Int
        val TEXTURE22: Int
        val TEXTURE23: Int
        val TEXTURE24: Int
        val TEXTURE25: Int
        val TEXTURE26: Int
        val TEXTURE27: Int
        val TEXTURE28: Int
        val TEXTURE29: Int
        val TEXTURE30: Int
        val TEXTURE31: Int
        val TEXTURE32: Int
        val TEXTURE33: Int
        val TEXTURE34: Int
        val TEXTURE35: Int
        val ACTIVE_TEXTURE: Int
        val REPEAT: Int
        val CLAMP_TO_EDGE: Int
        val MIRRORED_REPEAT: Int
        val FLOAT_VEC2: Int
        val FLOAT_VEC3: Int
        val FLOAT_VEC4: Int
        val INT_VEC2: Int
        val INT_VEC3: Int
        val INT_VEC4: Int
        val BOOL: Int
        val BOOL_VEC2: Int
        val BOOL_VEC3: Int
        val BOOL_VEC4: Int
        val SAMPLER_2D: Int
        val SAMPLER_CUBE: Int
        val VERTEX_ATTRIB_ARRAY_ENABLED:
Int\n val VERTEX_ATTRIB_ARRAY_SIZE: Int\n val VERTEX_ATTRIB_ARRAY_STRIDE: Int\n val VERTEX_ATTRIB_ARRAY_TYPE: Int\n val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int\n val VERTEX_ATTRIB_ARRAY_POINTER: Int\n val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int\n val IMPLEMENTATION_COLOR_READ_TYPE: Int\n val IMPLEMENTATION_COLOR_READ_FORMAT: Int\n val COMPILE_STATUS: Int\n val LOW_FLOAT: Int\n val MEDIUM_FLOAT: Int\n val HIGH_FLOAT: Int\n val LOW_INT: Int\n val MEDIUM_INT: Int\n val HIGH_INT: Int\n val FRAMEBUFFER: Int\n val RENDERBUFFER: Int\n val RGBA4: Int\n val RGB5_A1: Int\n val RGB565: Int\n val DEPTH_COMPONENT16: Int\n val STENCIL_INDEX: Int\n val STENCIL_INDEX8: Int\n val DEPTH_STENCIL: Int\n val RENDERBUFFER_WIDTH: Int\n val RENDERBUFFER_HEIGHT: Int\n val RENDERBUFFER_INTERNAL_FORMAT: Int\n val RENDERBUFFER_RED_SIZE: Int\n val RENDERBUFFER_GREEN_SIZE: Int\n val RENDERBUFFER_ALPHA_SIZE: Int\n val RENDERBUFFER_DEPTH_SIZE: Int\n val RENDERBUFFER_STENCIL_SIZE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int\n val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int\n val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int\n val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int\n val COLOR_ATTACHMENT0: Int\n val DEPTH_ATTACHMENT: Int\n val STENCIL_ATTACHMENT: Int\n val DEPTH_STENCIL_ATTACHMENT: Int\n val NONE: Int\n val FRAMEBUFFER_COMPLETE: Int\n val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int\n val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int\n val FRAMEBUFFER_UNSUPPORTED: Int\n val FRAMEBUFFER_BINDING: Int\n val RENDERBUFFER_BINDING: Int\n val MAX_RENDERBUFFER_SIZE: Int\n val INVALID_FRAMEBUFFER_OPERATION: Int\n val UNPACK_FLIP_Y_WEBGL: Int\n val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int\n val CONTEXT_LOST_WEBGL: Int\n val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int\n val BROWSER_DEFAULT_WEBGL: Int

/**
 * Exposes the JavaScript
 *[WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin
 */
public external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit = definedExternally) : Event {
    open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    constructor(eventInit: WebGLContextEventInit = definedExternally) : Event

    fun get() = definedExternally
    fun set(value: String) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["statusMessage"] = statusMessage
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 *[ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin
 */
public external open class ArrayBuffer(length: Int) : BufferDataSource {
    open val byteLength: Int
    fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer

    companion object {
        fun isView(value: Any?): Boolean
    }
}

/**
 * Exposes the JavaScript
 *[ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin
 */
public external interface ArrayBufferView : BufferDataSource {
    val buffer: ArrayBuffer
    val byteOffset: Int
    val byteLength: Int
    fun slice(begin: Int, end: Int = definedExternally): ArrayBufferView
}

/**
 * Exposes the JavaScript
 *[Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin
 */
public external class Int8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int)
}
Open Source Used In DNAC Platform Arica 9861

```kotlin
open class Uint32Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Uint32Array)
    constructor(array: Array<Int>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Uint32Array](https://developer.mozilla.org/en/docs/Web/API/Uint32Array) to Kotlin
 */
public external open class Uint32Array : ArrayBufferView {
    \n\n    constructor(length: Int)\n\n    constructor(array: Uint32Array)\n\n    constructor(array: Array<Int>)\n    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin
 */
public external open class Float32Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float32Array)
    constructor(array: Array<Float>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.get(index: Int): Float = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.set(index: Int, value: Float) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin
 */
public external open class Float64Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float64Array)
    constructor(array: Array<Double>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.get(index: Int): Double = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/ DataView) to Kotlin
 */
public external open class DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) : ArrayBufferView {
    \n\n    fun getInt8(byteOffset: Int): Byte
    fun getUint8(byteOffset: Int): Byte
    fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
    fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double
    fun getInt8(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint8(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
    fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double
    fun setInt8(byteOffset: Int, value: Byte) { asDynamic()[byteOffset] = value }
    fun setUint8(byteOffset: Int, value: Byte) { asDynamic()[byteOffset] = value }
    fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally): Int
    fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally): Int
    fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally): Int
    fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally): Int
    fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally): Int
    fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally): Int
    fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally): Float
    fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally): Double
}

public external interface BufferDataSource

public external interface TexImageSource
```

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 *
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.*/
Open Source Used In DNAC Platform Arica
Exposes the JavaScript
[CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin
*\n\n```kotlin
public abstract class CSSStyleDeclaration : ItemArrayLike<String> {
    open var cssText: String
    open val parentRule: CSSRule?
    open val cssFloat: String
    open var alignContent: String
    open var alignItems: String
    open var alignItems: String
    open var animationDelay: String
    open var animationDirection: String
    open var animationDuration: String
    open var animationFillMode: String
    open var animationIterationCount: String
    open var background: String
    open var backgroundAttachment: String
    open var backgroundClip: String
    open var backgroundOrigin: String
    open var backgroundPosition: String
    open var backgroundColor: String
    open var borderBottom: String
    open var borderBottomColor: String
    open var borderBottomLeftRadius: String
    open var borderBottomRightRadius: String
    open var borderBottomStyle: String
    open var borderBottomWidth: String
    open var borderCollapse: String
    open var borderColor: String
    open var borderImage: String
    open var borderImageOutset: String
    open var borderImageRepeat: String
    open var borderImageSlice: String
    open var borderTop: String
    open var borderTopColor: String
    open var borderTopLeftRadius: String
    open var borderTopRightRadius: String
    open var borderTopStyle: String
    open var borderTopWidth: String
    open var borderRadius: String
    open var borderRight: String
    open var borderRightColor: String
    open var borderRightStyle: String
    open var borderRightWidth: String
    open var borderSpacing: String
    open var borderTop: String
    open var borderTopColor: String
    open var borderTopLeftRadius: String
    open var borderTopRightRadius: String
    open var borderTopStyle: String
    open var borderTopWidth: String
    open var borderWidth: String
    open var bottom: String
    open var boxDecorationBreak: String
    open var boxShadow: String
    open var boxSizing: String
    open var breakAfter: String
    open var breakBefore: String
    open var breakInside: String
    open var breakBefore: String
    open var breakAfter: String
    open var breakInside: String
    open var captionSide: String
    open var clear: String
    open var clip: String
    open var color: String
    open var columnCount: String
    open var columnFill: String
    open var columnGap: String
    open var columnRule: String
    open var columnRuleColor: String
    open var columnRuleStyle: String
    open var columnRuleWidth: String
    open var columnSpan: String
    open var columnWidth: String
    open var columns: String
    open var content: String
    open var counterIncrement: String
    open var counterReset: String
    open var cursor: String
    open var direction: String
    open var emptyCells: String
    open var filter: String
    open var flex: String
    open var flexBasis: String
    open var flexGrow: String
    open var flexShrink: String
    open var flexWrap: String
    open var font: String
    open var fontFamily: String
    open var fontFeatureSettings: String
    open var fontKerning: String
    open var fontLanguageOverride: String
    open var fontsize: String
    open var fontSizeAdjust: String
    open var fontStyle: String
    open var fontSynthesis: String
    open var fontVariant: String
    open var fontVariantAlternates: String
    open var fontVariantEastAsian: String
    open var fontVariantLigatures: String
    open var fontVariantNumeric: String
    open var fontWeight: String
    open var hangingPunctuation: String
    open var height: String
    open var imageOrientation: String
    open var imageRendering: String
    open var imeMode: String
    open var justifyContent: String
    open var letterSpacing: String
    open var lineHeight: String
    open var listStyle: String
    open var listStylePosition: String
    open var listStyleType: String
    open var margin: String
    open var marginBottom: String
    open var marginRight: String
    open var marginTop: String
    open var mark: String
    open var markAfter: String
    open var marqueeDirection: String
    open var marqueePlayCount: String
    open var marqueeSpeed: String
    open var mask: String
    open var maskType: String
    open var maxHeigh: String
```
String
    open var maxWidth: String
    open var minHeight: String
    open var minWidth: String
    open var navDown: String
    open var navIndex: String
    open var navLeft: String
    open var navRight: String
    open var navUp: String
    open var objectFit: String
    open var objectPosition: String
    open var opacity: String
    open var order: String
    open var orphans: String
    open var outline: String
    open var outlineColor: String
    open var outlineOffset: String
    open var outlineStyle: String
    open var outlineWidth: String
    open var overflowWrap: String
    open var overflowX: String
    open var overflowY: String
    open var padding:
    open var paddingBottom: String
    open var paddingLeft: String
    open var paddingRight: String
    open var paddingTop: String
    open var pageBreakAfter: String
    open var pageBreakBefore: String
    open var pageBreakInside: String
    open var perspective: String
    open var perspectiveOrigin: String
    open var phonemes: String
    open var position: String
    open var quotes: String
    open var resize: String
    open var rest: String
    open var restAfter: String
    open var restBefore: String
    open var right: String
    open var tabSize: String
    open var tableLayout: String
    open var textAlign: String
    open var textAlignLast: String
    open var textCombineUpright: String
    open var textDecoration: String
    open var textDecorationColor: String
    open var textDecorationLine: String
    open var textDecorationStyle: String
    open var textIndent: String
    open var textJustify: String
    open var textOrientation: String
    open var textOverflow: String
    open var textShadow: String
    open var textTransform: String
    open var textUnderlinePosition: String
    open var top: String
    open var transform: String
    open var transformOrigin: String
    open var transformStyle: String
    open var transition: String
    open var transitionDelay: String
    open var transitionDuration: String
    open var transitionProperty: String
    open var transitionTimingFunction: String
    open var unicodeBidi: String
    open var verticalAlign: String
    open var visibility: String
    open var voiceBalance: String
    open var voicePitch: String
    open var voicePitchRange: String
    open var voiceRate: String
    open var voiceStress: String
    open var voiceVolume: String
    open var whiteSpace: String
    open var widows: String
    open var width: String
    open var wordBreak: String
    open var wordSpacing: String
    open var wordWrap: String
    open var writingMode: String
    open var zIndex: String
    open var _dashed_attribute: String
    open var _camel_cased_attribute: String
    open var _webkit_cased_attribute: String
    fun getPropertyValue(property: String): String
    fun getPriority(property: String): String
    fun setProperty(property: String, value: String, priority: String = definedExternally)
    fun setPropertyValue(property: String, value: String)
    fun setPropertyPriority(property: String, priority: String)
    fun removeProperty(property: String): String
    override fun item(index: Int): String
}

@kotlin.internal.InlineOnly
public inline operator fun CSSStyleDeclaration.get(index: Int): String? = asDynamic()[index]

public external interface ElementCSSInlineStyle {
    val style: CSSStyleDeclaration
}

/**
 * Exposes the JavaScript [CSS](https://developer.mozilla.org/en/docs/Web/API/CSS) to Kotlin
 */
public external abstract class CSS {...

companion object {
    fun escape(ident: String): String
}

package org.w3c.dom.encryptedmedia

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*

public external interface MediaKeySystemConfiguration {
    var label: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var initDataTypes: Array<String>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
    var videoCapabilities: ....

String
    open var max Width: String
    open var min Height: String
    open var min Width: String
    open var nav Down: String
    open var nav Index: String
    open var nav Left: String
    open var nav Right: String
    open var nav Up: String
    open var object Fit: String
    open var object Position: String
    open var opacity: String
    open var order: String
    open var orphans: String
    open var outline: String
    open var outline Color: String
    open var outline Offset: String
    open var outline Style: String
    open var outline Width: String
    open var overflow Wrap: String
    open var overflow X: String
    open var overflow Y: String
    open var padding:
    open var padding Bottom: String
    open var padding Left: String
    open var padding Right: String
    open var padding Top: String
    open var page Break After: String
    open var page Break Before: String
    open var page Break Inside: String
    open var perspective: String
    open var perspective Origin: String
    open var phonemes: String
    open var position: String
    open var quotes: String
    open var resize: String
    open var rest: String
    open var rest After: String
    open var rest Before: String
    open var right: String
    open var tab Size: String
    open var table Layout: String
    open var text Align: String
    open var text Align Last: String
    open var text Combine Upright: String
    open var text Decoration: String
    open var text Decoration Line: String
    open var text Decoration Style: String
    open var text Indent: String
    open var text Justify: String
    open var text Orientation: String
    open var text Overflow: String
    open var text Shadow: String
    open var text Transform: String
    open var text Underline Position: String
    open var top: String
    open var transform: String
    open var transform Origin: String
    open var transform Style: String
    open var transition: String
    open var transition Delay: String
    open var transition Duration: String
    open var transition Property: String
    open var transition Timing Function: String
    open var vertical Align: String
    open var visibility: String
    open var voice Balance: String
    open var voice Pitch: String
    open var voice Pitch Range: String
    open var voice Rate: String
    open var voice Stress: String
    open var voice Volume: String
    open var white Space: String
    open var widows: String
    open var width: String
    open var word Break: String
    open var word Spacing: String
    open var word Wrap: String
    open var writing Mode: String
    open var zIndex: String
    open var _dashed_attribute: String
    open var _camel_cased_attribute: String
    fun getPropertyValue(property: String): String
    fun getPriority(property: String): String
    fun setProperty(property: String, value: String, priority: String = definedExternally)
    fun setPropertyValue(property: String, value: String)
    fun setPropertyPriority(property: String, priority: String)
    fun removeProperty(property: String): String
    override fun item(index: Int): String
}

@kotlin.internal.InlineOnly
public inline operator fun CSSStyleDeclaration.get(index: Int): String? = asDynamic()[index]

public external interface ElementCSSInlineStyle {
    val style: CSSStyleDeclaration
}

/**
 * Exposes the JavaScript [CSS](https://developer.mozilla.org/en/docs/Web/API/CSS) to Kotlin
 */
public external abstract class CSS {...

companion object {
    fun escape(ident: String): String
}

package org.w3c.dom.encryptedmedia

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*

public external interface MediaKeySystemConfiguration {
    var label: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var initDataTypes: Array<String>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
    var audioCapabilities: ....
Array<MediaKeySystemMediaCapability>? /* = arrayOf() */
    get() = definedExternally
    set(value) = definedExternally

var distinctiveIdentifier: MediaKeysRequirement? /* = MediaKeysRequirement.OPTIONAL */
    get() = definedExternally
    set(value) = definedExternally

var persistentState: MediaKeysRequirement?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally

var sessionTypes: Array<String>?
    get() = definedExternally
    set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["label"] = label
    o["initDataTypes"] = initDataTypes
    o["audioCapabilities"] = audioCapabilities
    o["videoCapabilities"] = videoCapabilities
    o["distinctiveIdentifier"] = distinctiveIdentifier
    o["persistentState"] = persistentState
    o["sessionTypes"] = sessionTypes
    return o
}

public external interface MediaKeySystemMediaCapability {
    var contentType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var robustness: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaKeySystemMediaCapability(contentType: String? = "", robustness: String? = ""): MediaKeySystemMediaCapability {
    val o = js("({})")
    o["contentType"] = contentType
    o["robustness"] = robustness
    return o
}

public external abstract class MediaKeySystemAccess {
    open val keySystem: String
    fun getConfiguration(): MediaKeySystemConfiguration
    fun createMediaKeys(): Promise<MediaKeys>
}

public external abstract class MediaKeys {
    fun createSession(sessionType: MediaKeySessionType = definedExternally): MediaKeySession
}

public external abstract class MediaKeySession : EventTarget {
    open val sessionId: String
    open val expiration: Double
    open val closed: Promise<Unit>
    open var keyStatuses: MediaKeyStatusMap
    open var onKeyStatusesChange: ((Event) -> dynamic)?
    open var onMessage: ((MessageEvent) -> dynamic)?
    open var onKeyStatusesChange: ((Event) -> dynamic)?
    open var onMessage: ((MessageEvent) -> dynamic)?
    open val onKeyStatusesChange: ((Event) -> dynamic)?
    open val onMessage: ((MessageEvent) -> dynamic)?
    open val onKeyStatusesChange: ((Event) -> dynamic)?
    open val onMessage: ((MessageEvent) -> dynamic)?
    fun generateRequest(initDataType: String, initData: dynamic): Promise<Unit>
    fun load(sessionId: String): Promise<MediaKeySession>?
    fun remove(): Promise<Unit>
    fun update(response: dynamic): Promise<Unit>
    fun close(): Promise<Unit>
    fun remove(): Promise<Unit>
}

public external abstract class MediaKeyStatusMap {
    open val size: Int
    fun has(keyId: dynamic): Boolean
    fun get(keyId: dynamic): Any?
}

public external open class MediaKeyMessageEvent(type: String, eventInitDict: MediaKeyMessageEventInit) : Event {
    open val messageType: MediaKeyMessageType
    open val message: ArrayBuffer

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaKeyMessageEventInit : EventInit {
    var messageType: MediaKeyMessageType? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var message: ArrayBuffer?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["messageType"] = messageType
    o["message"] = message
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external open class MediaEncryptedEvent(type: String, eventInitDict: MediaEncryptedEventInit) : Event {
    open val initDataType: String
    open val initData: ArrayBuffer

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaEncryptedEventInit : EventInit {
    var initDataType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var initData: ArrayBuffer
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["initDataType"] = initDataType
    o["initData"] = initData
    return o
}
    val o = js("{}")
    o["initDataType"] = initDataType
    o["initData"] = initData
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
} /* please, don't implement this interface! */

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*

org.w3c.dom.encryptedmedia.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [UIEvent](https://developer.mozilla.org/en/docs/Web/API/UIEvent) to Kotlin
 */
public external open class UIEvent(type: String, eventInitDict: UIEventInit = definedExternally) : Event {
    open var view: Window?
    open val detail: Int

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface UIEventInit : EventInit {
    var view: Window? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var detail: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [FocusEvent](https://developer.mozilla.org/en/docs/Web/API/FocusEvent) to Kotlin
 */
public external open class FocusEvent(type: String, eventInitDict: FocusEventInit = definedExternally) : UIEvent {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface FocusEventInit : UIEventInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["relatedTarget"] = relatedTarget
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin
 */
public external open class MouseEvent(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent, UnionElementOrMouseEvent {
    open val screenX: Int
    open val screenY: Int
    open val clientX: Int
    open val clientY: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean
    open val region: String?
    open val pageX: Double
    open val pageY: Double
    open val x: Double
    open val y: Double
    open val offsetX: Double
    open val offsetY: Double
    open val button: Short
    open val buttons: Short
    fun getModifierState(keyArg: String): Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MouseEventInit : EventModifierInit {
    var screenX: Int? /* = 0 */
        get() = screenX
        set(value) = screenX
    var screenY: Int? /* = 0 */
        get() = screenY
        set(value) = screenY
    var clientX: Int? /* = 0 */
        get() = clientX
        set(value) = clientX
    var clientY: Int? /* = 0 */
        get() = clientY
        set(value) = clientY
    var button: Short? /* = 0 */
        get() = button
        set(value) = button
    var buttons: Short? /* = 0 */
        get() = buttons
        set(value) = buttons
    var relatedTarget: EventTarget? /* = null */
        get() = relatedTarget
        set(value) = relatedTarget
    var region: String? /* = null */
        get() = region
        set(value) = region
    var getModifierState(keyArg: String): Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}
    val o = js("({})")
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external interface EventModifierInit : UIEventInit {
    var ctrlKey: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierAltGraph: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierCapsLock: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierFn: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierHyper: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierNumLock: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierScrollLock: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierSuper: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierSymbol: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
    var modifierSymbolLock: Boolean? = false
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 [*WheelEvent*](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin
 */
public external open class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {
    open val deltaX: Double
    open val deltaY: Double
    open val deltaZ: Double
    open val deltaMode: Int

    companion object {
        val DOM_DELTA_PIXEL: Int
        val DOM_DELTA_LINE: Int
        val DOM_DELTA_PAGE: Int
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WheelEventInit : MouseEventInit {
    var deltaX: Double? = 0.0
        get() = definedExternally
        set(value) = definedExternally
    var deltaY: Double? = 0.0
        get() = definedExternally
        set(value) = definedExternally
    var deltaZ: Double? = 0.0
        get() = definedExternally
        set(value) = definedExternally
}
definedExternally
var deltaMode: Int? /* = 0 */

get() = definedExternally
set(value) =

definedExternally
@kotlin.internal.InlineOnly
public inline fun WheelEventInit(
    deltaX: Double? = 0.0,
    deltaY: Double? = 0.0,
    deltaMode: Int? = 0,
    screenX: Int? = 0,
    screenY: Int? = 0,
    clientX: Int? = 0,
    clientY: Int? = 0,
    button: Short? = 0,
    buttons: Short? = 0,
    relatedTarget: EventTarget? = null,
    region: String? = null,
    ctrlKey: Boolean? = false,
    shiftKey: Boolean? = false,
    altKey: Boolean? = false,
    metaKey: Boolean? = false,
    modifierAltGraph: Boolean? = false,
    modifierCapsLock: Boolean? = false,
    modifierFn: Boolean? = false,
    modifierFnLock: Boolean? = false,
    modifierHyper: Boolean? = false,
    modifierNumLock: Boolean? = false,
    modifierScrollLock: Boolean? = false,
    modifierSuper: Boolean? = false,
    modifierSymbol: Boolean? = false,
    modifierSymbolLock: Boolean? = false,
    view: Window? = null,
    detail: Int? = 0,
    bubbles: Boolean? = false,
    cancelable: Boolean? = false,
    composed: Boolean? = false): WheelEventInit {
    val o = js("({})")
    o["deltaX"] = deltaX
    o["deltaY"] = deltaY
    o["deltaZ"] = deltaZ
    o["deltaMode"] = deltaMode
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return
}

/**
 * Exposes the JavaScript [InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to
 * Kotlin
 */
public external open class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) :
    UIEvent {
    open val data: String
    open val isComposing: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface InputEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun InputEventInit(data: String? = "", isComposing: Boolean? = false,
    view: Window? = null,
    detail: Int? = 0,
    bubbles: Boolean? = false,
    cancelable: Boolean? = false,
    composed: Boolean? = false): InputEventInit {
    val o = js("({})")
    o["data"] = data
    o["isComposing"] = isComposing
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return
}

/**
 * Exposes the JavaScript [KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin
 */
public external open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) :
    UIEvent {
    open val key: String
    open val code: String
    open val location: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean
    open val repeat: Boolean

    fun getModifierState(keyArg: String): Boolean

    companion object {
        val DOM_KEY_LOCATION_STANDARD: Int
        val DOM_KEY_LOCATION_LEFT: Int
        val DOM_KEY_LOCATION_RIGHT: Int
        val DOM_KEY_LOCATION_NUMPAD: Int
    }
}

public external interface KeyboardEventInit : EventModifierInit {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var data: String
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean?
        get() = definedExternally
    var repeat: Boolean?
        get() = definedExternally
    var value: String
        get() = definedExternally
    var isComposing: Boolean
        get() = definedExternally
    var relatedTarget: EventTarget
        get() = definedExternally
        set(value) = definedExternally
    var view: Window
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun KeyboardEventInit(key: String? = "\"",
    code: String? = "\"",
    location: Int? = 0,
    repeat: Boolean? = false,
    isComposing: Boolean? = false,
    value: String? = "\"",
    relatedTarget: EventTarget? = null,
    view: Window? = null,
    detail: Int? = 0,
    bubbles: Boolean? = false,
    cancelable: Boolean? = false,
    composed: Boolean? = false): KeyboardEventInit {
    val o = js("({})")
    o["key"] = key
    o["code"] = code
    o["location"] = location
    o["repeat"] = repeat
    o["isComposing"] = isComposing
    o["value"] = value
    o["relatedTarget"] = relatedTarget
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return
}
    val o =
    o["key"] = key
    o["code"] = code
    o["location"] = location
    o["repeat"] = repeat
    o["isComposing"] = isComposing
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin
 */
public external open class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) :
    UIEvent {
    open val data: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CompositionEventInit : UIEventInit {
    var data: String? /* = "" */
        get() =
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CompositionEventInit(data: String? = "", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
cancelable: Boolean? = false, composed: Boolean? = false): CompositionEventInit {
    val o =
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin
 */
public external open class Event(type: String, eventInitDict: EventInit = definedExternally) {
    open val type: String
    open val target: EventTarget?
    open val currentTarget: EventTarget?
    open val eventPhase: Short
    open val bubbles: Boolean
    open val cancelable: Boolean
    open val defaultPrevented: Boolean
    open val composed: Boolean
    open val isTrusted: Boolean
    open val timeStamp: Number
    fun composedPath(): Array<EventTarget>
    fun stopPropagation(): Boolean
    fun stopImmediatePropagation(): Boolean
    fun preventDefault(): Boolean
    fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean): Boolean
}

/**
 * Exposes the JavaScript [EventListener](https://developer.mozilla.org/en/docs/Web/API/EventListener) to Kotlin
 */
public external interface EventListener {
    fun handleEvent(event: Event)
}
org.w3c.dom.pointerevents.*
uniport org.w3c.dom.svg.*
uniport org.w3c.dom.url.*
uniport org.w3c.fetch.*
uniport org.w3c.files.*
uniport org.w3c.notifications.*
uniport org.w3c.performance.*
uniport org.w3c.workers.*
uniport org.w3c.xhr.*

public external abstract class HTMLAllCollection { 
    open val length: Int
    fun item(nameOrIndex: String = definedExternally): UnionElementOrHTMLCollection? 
    fun namedItem(name: String): UnionElementOrHTMLCollection? 
    @kotlin.internal.InlineOnly 
    public inline operator fun HTMLAllCollection.get(index: Int): Element? = 
        asDynamic()[index]@kotlin.internal.InlineOnly
    
    /** 
     * Exposes the JavaScript [HTMLFormControlsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLFormControlsCollection) to Kotlin
     */
    public external abstract class HTMLFormControlsCollection : HTMLCollection

    /** 
     * Exposes the JavaScript [RadioNodeList](https://developer.mozilla.org/en/docs/Web/API/RadioNodeList) to Kotlin
     */
    public external abstract class RadioNodeList : NodeList, UnionElementOrRadioNodeList {
        open var value: String
    }

    /** 
     * Exposes the JavaScript [HTMLOptionsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionsCollection) to Kotlin
     */
    public external abstract class HTMLOptionsCollection : HTMLCollection {
        override var length: Int
        open var selectedIndex: Int
        fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
        fun remove(index: Int)
        @kotlin.internal.InlineOnly 
        public inline operator fun HTMLOptionsCollection.set(index: Int, option: HTMLOptionElement?) { asDynamic()[index] = option 
    }

    /** 
     * Exposes the JavaScript [HTMLElement](https://developer.mozilla.org/en/docs/Web/API/HTMLElement) to Kotlin
     */
    public external abstract class HTMLElement : Element, GlobalEventHandlers, DocumentAndElementEventHandlers, 
        ElementContentEditable, ElementCSSInlineStyle { 
        open var title: String
        open var lang: String
        open var translate: Boolean
        open var dir: String
        open val dataset: DOMStringMap
        open var hidden: Boolean
        open var tabIndex: Int
        open var accessKey: String
        open val accessKeyLabel: String
        open var draggable: Boolean
        open val dropzone: DOMTokenList
        open var contextMenu: HTMLMenuElement?
        open var spellcheck: Boolean
        open var innerText: String
        open val offsetParent: Element?
        open val offsetTop: Int
        open val offsetLeft: Int
        open val offsetWidth: Int
        open val offsetHeight: Int
        fun click()
        fun focus()
        fun blur()
        fun forceSpellCheck()

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }

    /** 
     * Exposes the JavaScript [HTMLUnknownElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUnknownElement) to Kotlin
     */
    public external abstract class HTMLUnknownElement : HTMLElement {
        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }
DOMStringMap

**DOMStringMap**

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.get(name: String): String? = asDynamic()[name]

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.set(name: String, value: String) { asDynamic()[name] = value }
```

**HTMLHeadElement**

```kotlin
public external abstract class HTMLHeadElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

**HTMLTitleElement**

```kotlin
public external abstract class HTMLTitleElement : HTMLElement {
    open var text: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

**HTMLBaseElement**

```kotlin
public external abstract class HTMLBaseElement : HTMLElement {
    open var href: String
    open var target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```
Open Source Used In DNAC Platform Arica 9874

[HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin

```kotlin
public external abstract class HTMLLinkElement : HTMLElement, LinkStyle {
    open var href: String
    open var crossOrigin: String?
    open var rel: String
    open var `as`: RequestDestination
    open val relList: DOMTokenList
    open var media: String
    open var nonce: String
    open var hreflang: String
    open var type: String
    open val sizes: DOMTokenList
    open var referrerPolicy: String
    open var charset: String
    open var rev: String
    open var target: String
    open var scope: String
    open var workerType: WorkerType

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

[HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin

```kotlin
public external abstract class HTMLMetaElement : HTMLElement {
    open var name: String
    open var httpEquiv: String
    open var content: String
    open var scheme: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

[HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin

```kotlin
public external abstract class HTMLStyleElement : HTMLElement, LinkStyle {
    open var media: String
    open var nonce: String
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

[HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin

```kotlin
public external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {
    open var text: String
    open var link: String
    open var vLink: String
    open var aLink: String
    open var bgColor: String
    open var background: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```
DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin

```
public external abstract class HTMLHeadingElement : HTMLElement {
    open var align: String
    open var noShade: Boolean
    open var color: String
    open var size: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

Exposes the JavaScript [HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin

```
public external abstract class HTMLParagraphElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

Exposes the JavaScript [HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin

```
public external abstract class HTMLHRElement : HTMLElement {
    open var align: String
    open var color: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

Exposes the JavaScript [HTMLPreElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPreElement) to Kotlin

```
public external abstract class HTMLPreElement : HTMLElement {
    open var width: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```
/* Exposes the JavaScript HTMLQuoteElement to Kotlin */
public external abstract class HTMLQuoteElement : HTMLElement {
    open var cite: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript HTMLOListElement to Kotlin */
public external abstract class HTMLOListElement : HTMLElement {
    open var reversed: Boolean
    open var start: Int
    open var type: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript HTMLUListElement to Kotlin */
public external abstract class HTMLUListElement : HTMLElement {
    open var compact: Boolean
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript HTMLLIElement to Kotlin */
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript HTMLDListElement to Kotlin */
public external abstract class HTMLDListElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
external abstract class HTMLDListElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin
 */
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin
 */
public external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var hreflang: String
    open var type: String
    open var text: String
    open var referrerPolicy: String
    open var coords: String
    open var charset: String
    open var name: String
    open var rev: String
    open var shape: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDataElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataElement) to Kotlin
 */
public external abstract class HTMLDataElement : HTMLElement {
    open var value: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin
 */
public
external abstract class HTMLTimeElement : HTMLElement {
    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin */
public external abstract class HTMLSpanElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin */
public external abstract class HTMLBRElement : HTMLElement {
    open var clear: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to Kotlin */
public external interface HTMLHyperlinkElementUtils {
    var href: String
    val origin: String
    var protocol: String
    var username: String
    var password: String
    var host: String
    var hostname: String
    var port: String
    var pathname: String
    var search: String
    var hash: String
}

/**
 * Exposes the JavaScript [HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin */
public external abstract class HTMLModElement : HTMLElement {
    open var cite: String
    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
private external abstract class HTMLPictureElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class HTMLSourceElement : HTMLElement {
    open var src: String
    open var type: String
    open var srcset: String
    open var sizes: String
    open var media: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {
    open var alt: String
    open var src: String
    open var srcset: String
    open var sizes: String
    open var crossOrigin: String?
    open var useMap: String
    open var isMap: Boolean
    open var width: Int
    open var height: Int
    open val naturalWidth: Int
    open val naturalHeight: Int
    open val complete: Boolean
    open var lowsrc: String
    open var align: String
    open var hspace: Int
    open var vspace: Int
    open var longDesc: String
    open var border: String
    open val x: Int
    open val y: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String
    open var srcdoc: String
    open var name: String
    open val sandbox: DOMTokenList
    open var allowFullscreen: Boolean
    open var allowUserMedia: Boolean
    open var width: String
    open var height: String
    open var referrerPolicy: String
    open var referrer: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var align: String
    open var scrolling: String
    open var frameborder: String
    open var longDesc: String
    open var marginHeight: String
    open var marginWidth: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/* Exposes the JavaScript [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin

```kotlin
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String
    open var type: String
    open var width: String
    open var height: String
    open var align: String
    open var name: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

/* Exposes the JavaScript [HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin

```kotlin
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String
    open var type: String
    open var typeMustMatch: Boolean
    open var name: String
    open var useMap: String
    open var form: HTMLFormElement?
    open var width: String
    open var height: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var align: String
    open var archive: String
    open var code: String
    open var declare: Boolean
    open var hspace: Int
    open var standby: String
    open var vspace: Int
    open var codeBase: String
    open var codeType: String
    open var border: String
    fun getSVGDocument(): Document?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

/* Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin

```kotlin
public external abstract class HTMLParamElement : HTMLElement {
    open var name: String
    open var value: String
    open var type: String
    open var valueType: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
    }
```
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENTPOSITION_DISCONNECTED: Short
Open Source Used In DNAC Platform Arica

```kotlin
public abstract class HTMLMediaElement : HTMLElement {
    open val error: MediaError?
    open var src: String
        open var srcObject: MediaProvider?
    open val currentSrc: String
    open var crossOrigin: String?
    open val networkState: Short
    open var preload: String
    open val buffered: TimeRanges
    open val readyState: Short
    open var seeking: Boolean
    open var currentTime: Double
    open val duration: Double
    open var autoplay: Boolean
    open var loop: Boolean
    open var controls: Boolean
    open var volume: Double
    open var muted: Boolean
    open var defaultMuted: Boolean
    open val audioTracks: AudioTrackList
    open val videoTracks: VideoTrackList
    open val textTracks: TextTrackList
    open val mediaKeys: MediaKeys?
    open var onencrypted: ((Event) -> dynamic)?
    open var onwaitingforkey: ((Event) -> dynamic)?
    fun load()
    fun canPlayType(type: String): CanPlayTypeResult
    fun fastSeek(time: Double)
    fun getStartDate(): dynamic
    fun play(): Promise<Unit>
    fun pause()
    fun addTextTrack(kind: TextTrackKind, label: String = definedExternally, language: String = definedExternally): TextTrack
    fun setMediaKeys(mediaKeys: MediaKeys?): Promise<Unit>

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin
 */
public abstract class MediaError {
    open val code: Short

    companion object {
        val MEDIA_ERR_ABORTED: Short
        val MEDIA_ERR_NETWORK: Short
        val MEDIA_ERR_DECODE: Short
        val MEDIA_ERR_SRC_NOT_SUPPORTED: Short
    }
}

/**
 * Exposes the JavaScript [AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin
 */
public abstract class AudioTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): AudioTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun AudioTrackList.get(index: Int): AudioTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [AudioTrack](https://developer.mozilla.org/en/docs/Web/API/AudioTrack) to Kotlin
 */
public abstract class AudioTrack : UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind:
    open val label: String
    open val language: String
    open var enabled: Boolean
    open val sourceBuffer: SourceBuffer?
}

/**
 * Exposes the JavaScript [VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin
 */
public abstract class VideoTrackList : EventTarget {
    open val length: Int
    open val selectedIndex: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    fun getTrackByIndex(index: Int): VideoTrack?
}
```

Open Source Used In DNAC Platform Arica
The code snippet provided seems to be a representation of the JavaScript API for text tracks in a web browser's media player. It includes various classes and methods related to handling text tracks, including methods for adding, removing, and manipulating text tracks and cues. The code is written in Kotlin and includes inline XML annotations for JavaScript events.

The snippet contains classes such as `TextTrack`, `TextTrackList`, `TextTrackCueList`, `TextTrackCue`, and `TimeRanges`, each with their respective properties and methods. The `TextTrack` class, for example, contains properties like `kind`, `label`, `language`, `id`, `inBandMetadataTrackDispatchType`, and `cues`, along with methods for adding and removing cues.

The code is part of a larger framework that interacts with the browser's media player API, allowing developers to dynamically manipulate text tracks in a video or audio content.

```
public external abstract class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val kind: TextTrackKind
    open val label: String
    open val language: String
    open val id: String
    open val inBandMetadataTrackDispatchType: String
    open var mode: TextTrackMode
    open val cues: TextTrackCueList?
    open val activeCues: TextTrackCueList?
    open var oncuechange: ((Event) -> dynamic)?
    open val sourceBuffer: SourceBuffer?
    fun addCue(cue: TextTrackCue)
    fun removeCue(cue: TextTrackCue)
}
```

This is just a snippet of the larger codebase, which is designed to provide a Kotlin interface to the JavaScript API for text tracks, enabling developers to write more idiomatic Kotlin code for managing media player features.
referrerPolicy: String
    open var noHref: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var caption: HTMLTableCaptionElement?
    open var tHead: HTMLTableSectionElement?
    open var tFoot: HTMLTableSectionElement?
    open val tBodies: HTMLCollection
    open val rows: HTMLCollection
    open var align: String
    open var border: String
    open var frame: String
    open var rules: String
    open var summary: String
    open var width: String
    open var bgColor: String
    open var cellPadding: String
    open var cellSpacing: String

    fun createCaption(): HTMLTableCaptionElement
    fun deleteCaption()
    fun createTHead(): HTMLTableSectionElement
    fun deleteTHead()
    fun createTFoot(): HTMLTableSectionElement
    fun deleteTFoot()
    fun createTBody(): HTMLTableSectionElement
    fun insertRow(index: Int = definedExternally): HTMLTableRowElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin
 */
public external abstract class HTMLTableCaptionElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin
 */
public external abstract class HTMLTableColElement : HTMLElement {
    open var span: Int
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

*/

/**
 * Exposes the JavaScript [HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin
 */
public external abstract class HTMLTableSectionElement : HTMLElement {
    open val rows: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    fun insertRow(index: Int = definedExternally): HTMLElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin
 */
public external abstract class HTMLTableRowElement : HTMLElement {
    open val rowIndex: Int
    open var sectionRowIndex: Int
    open val cells: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var bgColor: String
    fun insertCell(index: Int = definedExternally): HTMLElement
    fun deleteCell(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin
 */
public external abstract class HTMLTableCellElement : HTMLElement {
    open var colSpan: Int
    open var rowSpan: Int
    open var headers: String
    open val cellIndex: Int
    open var scope: String
    open var abbr: String
    open var align: String
    open var axis: String
    open var height: String
    open var width: String
    open var ch: String
    open var chOff: String
    open var noWrap: Boolean
    open var vAlign: String
    open var bgColor: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin
 */
public external abstract class HTMLFormElement : HTMLElement {
    open var acceptCharset: String
    open var
action: String
open var autocomplete: String
open var enctype: String
open var encoding: String
open var method: String
open var name: String
open var noValidate: Boolean
open var target: String
open var elements: HTMLFormControlsCollection
open val length: Int
fun submit(): Unit
fun reset(): Unit
fun checkValidity(): Boolean
fun reportValidity(): Boolean
companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(name: String): UnionElementOrRadioNodeList? = asDynamic()[name]

/**
 * Exposes the JavaScript [HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin
 */
public external abstract class HTMLLabelElement : HTMLElement {
open val form: HTMLFormElement?
open var htmlFor: String
open val control: HTMLElement?

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin
 */
public external abstract class HTMLInputElement : HTMLElement {
open var accept: String
open var alt: String
open var autocomplete: String
open var autofocus: Boolean
open var defaultChecked: Boolean
open var checked: Boolean
open var dirName: String
open var disabled: Boolean
open val form: HTMLFormElement?
open val files: FileList?
open var formAction: String
open var formEnctype: String
open var formMethod: String
open var formNoValidate: Boolean
open var formTarget: String
open var height: Int
open var indeterminate: Boolean
open var inputMode: String
open var label: String
open var type: String
open var valueAsDate: dynamic
open var valueAsNumber: Double
open var width: Int
open var willValidate: Boolean
open var validity: ValidityState
open var validationMessage: String
open val elements: HTMLFormControlsCollection
open val length: Int
fun submit(): Unit
fun reset(): Unit
fun checkValidity(): Boolean
fun reportValidity(): Boolean
companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

\n
Short

val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin
 */
public external abstract class HTMLButtonElement : HTMLElement {
    open var autofocus: Boolean
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var name: String
    open var type: String
    open var value: String
    open var menu: HTMLMenuElement?
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin
 */
public external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {
    open var autocomplete: String
    open var autofocus: Boolean
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var multiple: Boolean
    open var name: String
    open var required: Boolean
    open var size: Int
    open var type: String
    open val options: HTMLOptionsCollection
    override var length: Int
    open val selectedOptions: HTMLCollection
    open var selectedIndex: Int
    open var value: String
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun namedItem(name: String): HTMLOptionElement?
    fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
    fun remove(index: Int)
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    override fun item(index: Int): Element?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

    @kotlin.internal.InlineOnly
    public inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]
    @kotlin.internal.InlineOnly
    public inline operator fun HTMLSelectElement.set(index: Int,
Open Source Used In DNAC Platform Arica 9888

```kotlin
/**
 * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin
 */
public external abstract class HTMLDataListElement : HTMLElement {
    open val options: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin
 */
public external abstract class HTMLOptGroupElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
    open var disabled: Boolean
    open var label: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionElement) to Kotlin
 */
public external abstract class HTMLOptionElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var label: String
    open var defaultSelected: Boolean
    open var selected: Boolean
    open var value: String
    open var text: String
    open val index: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement) to Kotlin
 */
public external abstract class HTMLTextAreaElement : HTMLElement {
    open var autocomplete: String
    open var autofocus: Boolean
    open var cols: Int
    open var dirName: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var inputMode: String
    open var maxLength: Int
    open var minLength: Int
    open var name: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var rows: Int
    open var wrap: String
    open var type: String
    open var defaultValue: String
    open var value: String
    open val textLength: Int
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    open var selectionStart: Int?
    open var selectionEnd: Int?
    open var selectionDirection: String?
    fun checkValidity():
```
fun reportValidity(): Boolean
fun setCustomValidity(error: String)
fun select()
fun setRangeText(replacement: String)
fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement) to Kotlin
 */
public external abstract class HTMLKeygenElement : HTMLElement {
    open var autofocus: Boolean
    open var challenge: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var keytype: String
    open var name: String
    open val type: String
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin
 */
public external abstract class HTMLOutputElement : HTMLElement {
    open val htmlFor: DOMTokenList
    open val form: HTMLFormElement?
    open var name: String
    open val type: String
    open var defaultValue: String
    open var value: String
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin
 */
public external abstract class HTMLProgressElement : HTMLElement {
    open var value: Double
    open var max: Double
    open val position: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
**Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin**

```kotlin
public external abstract class HTMLMeterElement : HTMLElement {
    open var value: Double
    open var min: Double
    open var max: Double
    open var low: Double
    open var high: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin
 */
public external abstract class HTMLFieldSetElement : HTMLElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var name: String
    open val type: String
    open val elements: HTMLCollection
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin
 */
public external abstract class HTMLLegendElement : HTMLElement {
    open val form: HTMLFormElement?
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin
 */
public external
abstract class ValidityState {
    open val valueMissing: Boolean
    open val typeMismatch: Boolean
    open val
    patternMismatch: Boolean
    open val tooLong: Boolean
    open val tooShort: Boolean
    open val rangeUnderflow: Boolean
    open val rangeOverflow: Boolean
    open val stepMismatch: Boolean
    open val badInput: Boolean
    open val customError: Boolean
    open val valid: Boolean
}

/**
 * Exposes the
 * JavaScript [HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to
 * Kotlin
 */
public external abstract class HTMLDetailsElement : HTMLElement {
    open var open: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external abstract class HTMLMenuElement : HTMLElement {
    open var type: String
    open var label: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external abstract class HTMLMenuItemElement : HTMLElement {
    open var type: String
    open var label: String
    open var icon: String
    open var disabled: Boolean
    open var checked: Boolean
    open var radiogroup: String
    open var default: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external open class RelatedEvent(type: String, eventInitDict: RelatedEventInit = definedExternally) : Event {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["relatedTarget"] = relatedTarget
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the
 * JavaScript [HTMLDialogElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDialogElement) to
 * Kotlin
 */
*public external abstract class HTMLDialogElement : HTMLElement {
    open var open: Boolean
    open var returnValue: String
    fun show(anchor: UnionElementOrMouseEvent = definedExternally)
definedExternally
    set(value) = definedExternally
    var role: String? /* = null */
    get() =
    definedExternally
    set(value) = definedExternally

    val o = js("()")
    o["path"] = path
    o["fillRule"] = fillRule
    o["id"] = id
    o["parentID"] = parentID
    o["cursor"] = cursor
    o["control"] = control
    o["label"] = label
    o["role"] = role
    return o
}

@kotlin.internal.InlineOnly
public inline fun
    val o = js("()")
    o["path"] = path
    o["fillRule"] = fillRule
    o["id"] = id
    o["parentID"] = parentID
    o["cursor"] = cursor
    o["control"] = control
    o["label"] = label
    o["role"] = role
    return o
}

/**
 * Exposes the JavaScript
 */
public external open class
ImageData : ImageBitmapSource, TexImageSource {
    constructor(sw: Int, sh: Int)
    constructor(data: Uint8ClampedArray, sw: Int, sh: Int)
    open val width: Int
    open val height: Int
    open val data: Uint8ClampedArray
}

/**
 * Exposes the JavaScript
 */
public external open class
Path2D() : CanvasPath {
    constructor(path: Path2D)
    constructor(paths: Array<Path2D>, fillRule: CanvasFillRule = definedExternally)
    constructor(d: String)
    fun addPath(path: Path2D, transform: dynamic = definedExternally)
    override fun closePath()
    override fun moveTo(x: Double, y: Double)
    override fun lineTo(x: Double, y: Double)
    override fun quadraticCurveTo(cp1x: Double, cp1y: Double, x: Double, y: Double)
    override fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
    override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
    override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double)
    override fun rect(x: Double, y: Double, w: Double, h: Double)
    override fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)
    override fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)
}

/**
 * Exposes the JavaScript
 */
public external abstract class
ImageBitmapRenderingContext {
    open val canvas: HTMLCanvasElement
    fun transferFromImageBitmap(bitmap: ImageBitmap?)
}

public external interface
ImageBitmapRenderingContextSettings {
    var alpha: Boolean? /* = true */
    get() = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun
    val o = js("()")
    o["alpha"] = alpha
    return o
}

/**
 * Exposes the JavaScript
 */
public external abstract class
CustomElementRegistry {
    fun define(name: String, constructor: () -> dynamic, options: ElementDefinitionOptions = definedExternally)
    fun get(name: String): Any?
    fun whenDefined(name: String): Promise<Unit>
}

public external interface
ElementDefinitionOptions {
    var extends: String?
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun
ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {
    val o = js("()")
    o["extends"] = extends
    return o
}

public external interface
ElementContentEditable {
    var contentEditable: String
    val isContentEditable: Boolean
}

/**
 * Exposes the JavaScript
 */
public external abstract class
DataTransfer {
    open var dropEffect: String
    open var effectAllowed: String
    open val items: DataTransferItemList
    open val types: Array<out String>
    open val files: FileList
    fun setDragImage(image: Element, x: Int, y: Int)
    fun getData(format: String): String
    fun setData(format: String, data: String)
    fun clearData(format: String): DataTransferItemList
    fun open var dropEffect: String
    fun open var effectAllowed: Allowed: String
    fun open val items: Array<String>
    fun open val types: Array<out String>
    fun open val files: FileList
    fun setDragImage(image: Element, x: Int, y: Int)
    fun getData(format: String): String
    fun setData(format: String, data: String)
    fun clearData(format: String): DataTransferItemList
}

public external abstract class
DataTransferItemList {
    open val length: Int
    fun add(data: String): DataTransferItem
    fun add(data: File): DataTransferItem
    fun remove(index: Int)
    fun clear(): Unit
}

Open Source Used In DNAC Platform Arica 9895
DataTransferItem? = asDynamic()[index]\n\n/**
 * Exposes the JavaScript
 [DataTransferItem](https://developer.mozilla.org/en/docs/Web/API/DataTransferItem) to Kotlin
 */

external abstract class DataTransferItem {
    open val kind: String
    open val type: String
    fun getAsString(_callback: ((String) -> Unit)?)
    fun getAsFile(): File?
}

/**
 * Exposes the JavaScript
 [DragEvent](https://developer.mozilla.org/en/docs/Web/API/DragEvent) to Kotlin
 */
public external open class DragEvent(type: String, eventInitDict: DragEventInit = definedExternally) : MouseEvent {
    open val dataTransfer: DataTransfer?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface DragEventInit : MouseEventInit {
        var dataTransfer: DataTransfer? /* = null */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["dataTransfer"] = dataTransfer
        o["screenX"] = screenX
        o["screenY"] = screenY
        o["clientX"] = clientX
        o["clientY"] = clientY
        o["button"] = button
        o["relatedTarget"] = relatedTarget
        o["region"] = region
        o["ctrlKey"] = ctrlKey
        o["shiftKey"] = shiftKey
        o["metaKey"] = metaKey
        o["altKey"] = altKey
        o["dataTransfer"] = dataTransfer
        o["dataTransfer"] = dataTransfer
        o["dataTransfer"] = dataTransfer
        o["dataTransfer"] = dataTransfer
    }

    /**
     * Exposes the JavaScript
     [Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin
     */
    public external abstract class Window : EventTarget, GlobalEventHandlers, WindowEventHandlers, WindowOrWorkerGlobalScope, WindowSessionStorage, WindowLocalStorage, GlobalPerformance, UnionMessagePortOrWindowProxy {
        open val window: Window
        open val self: Window
        open val document: Document
        open var name: String
        open val location: Location
        open val history: History
        open val customElements: CustomElementRegistry
        open val locationbar: BarProp
        open val menubar: BarProp
        open val personalbar: BarProp
        open val scrollbars: BarProp
        open val statusbar: BarProp
        open val toolbar: BarProp
        open var status: String
        open val closed: Boolean
        open val frames: Window
        open val length: Int
        open val top: Window
        open var opener: Any?
        open val parent: Window
        open val frameElement: Element
        open val navigator: Navigator
        open val applicationCache: ApplicationCache
        open val external: External
        open val screen: Screen
        open val innerWidth: Int
        open val innerHeight: Int
        open val scrollX: Double
        open val scrollY: Double
        open val pageXOffset: Double
        open val screenX: Int
        open val screenY: Int
        open val outerWidth: Int
        open val outerHeight: Int
        open val devicePixelRatio: Double
        fun close(): Unit
        fun focus(): Unit
        fun blur(): Unit
        fun open(url: String = definedExternally, target: String = definedExternally, features: String = definedExternally): Window
        fun alert(): Unit
        fun alert(message: String)
        fun confirm(message: String): Boolean
        fun prompt(message: String = definedExternally, default: String = definedExternally): String?
        fun postMessage(message: Any?, targetOrigin: String, transfer: Array<dynamic> = definedExternally)
        fun captureEvents(): Unit
        fun releaseEvents(): Unit
        fun matchMedia(query: String): MediaQueryList
        fun moveTo(x: Int, y: Int)
        fun moveBy(x: Int, y: Int)
        fun resizeTo(x: Int, y: Int)
        fun resizeBy(x: Int, y: Int)
        fun resizeTo(x: Int, y: Int)
        fun resizeBy(x: Int, y: Int)
    }
get() = definedExternally
set(value) = definedExternally
var oncuechange: ((Event) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondblclick: ((MouseEvnet) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondrag: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondragend: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondragexit: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondragover: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondragstart: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondrop: ((DragEvent) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var ondurationchange: ((Event) -> dynamic)?

get() = definedExternally
set(value) = definedExternally
var onerror: ((dynamic, String, Int, Int, Any?) ->
        dynamic)?

get() = definedExternally
set(value) = definedExternally
var onfocus: ((FocusEvent) ->
        dynamic)?

get() = definedExternally
set(value) = definedExternally
var oninput: ((InputEvent) ->
        dynamic)?

get() = definedExternally
set(value) = definedExternally
var oninvalid: ((Event) ->
        dynamic)?

get() = definedExternally
set(value) = definedExternally
var onkeydown:

get() = definedExternally
set(value) = definedExternally
var onkeypress:

get() = definedExternally
set(value) = definedExternally
var onkeyup:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var oncontextmenu:

get() = definedExternally
set(value) = definedExternally
var onclick:

get() = definedExternally
set(value) = definedExternally
var onchange:

get() = definedExternally
set(value) = definedExternally
var onbeforeunload:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var oncontextmenu:

get() = definedExternally
set(value) = definedExternally
var onclick:

get() = definedExternally
set(value) = definedExternally
var onchange:

get() = definedExternally
set(value) = definedExternally
var onbeforeunload:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var onblur:

get() = definedExternally
set(value) = definedExternally
var oncontextmenu:

get() = definedExternally
set(value) = definedExternally
var onclick:

get() = definedExternally
set(value) = definedExternally
var onchange:

get() = definedExternally
set(value) = definedExternally
var onbeforeunload:
Open Source Used In DNAC Platform Arica 9901

Promise<ImageBitmap>

fun fetch(input: dynamic, init: RequestInit = definedExternally):
Promise<Response>

/**
 * Exposes the JavaScript
[Navigator](https://developer.mozilla.org/en/docs/Web/API/Navigator) to Kotlin
 */
public external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {
    open val open: val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val allowedMediaTypes: MediaMediaTypes
    open val requestMediaKeySystemAccess(keySystem: String, supportedConfigurations: Array<MediaKeySystemConfiguration>): Promise<MediaKeySystemAccess>
    fun requestMediaKeySystemAccess(keySystem: String, supportedConfigurations: Array<MediaKeySystemConfiguration>): Promise<MediaKeySystemAccess>
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

/**
 * Exposes the JavaScript
[NavigatorID](https://developer.mozilla.org/en/docs/Web/API/NavigatorID) to Kotlin
 */
public external interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val platform: String
    val product: String
    val productSub: String
    val userAgent: String
    val vendor: String
    val vendorSub: String
    fun taintEnabled(): Boolean
}

/**
 * Exposes the JavaScript
[NavigatorLanguage](https://developer.mozilla.org/en/docs/Web/API/NavigatorLanguage) to Kotlin
 */
public external interface NavigatorLanguage {
    val language: String
    val languages: Array<out String>
}

public external interface NavigatorContentUtils {
    fun registerProtocolHandler(scheme: String, url: String, title: String)
    fun registerContentHandler(mimeType: String, url: String, title: String)
    fun isProtocolHandlerRegistered(scheme: String, url: String): String
    fun isContentHandlerRegistered(mimeType: String, url: String): String
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String, title: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

/**
 * Exposes the JavaScript
[NavigatorPlugins](https://developer.mozilla.org/en/docs/Web/API/NavigatorPlugins) to Kotlin
 */
public external interface NavigatorPlugins {
    val plugins: PluginArray
    val mimeTypes: MimeTypeArray
    fun javaEnabled(): Boolean
}

/**
 * Exposes the JavaScript
[PluginArray](https://developer.mozilla.org/en/docs/Web/API/PluginArray) to Kotlin
 */
public external abstract class PluginArray : ItemArrayLike<Plugin> {
    fun refresh(reload: Boolean = definedExternally)
    override fun item(index: Int): Plugin?
    fun namedItem(name: String): Plugin?
}

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(index: Int): Plugin? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]

/**
 * Exposes the JavaScript
[MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin
 */
public abstract class MimeTypeArray : ItemArrayLike<MimeType> {
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
}

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(index: Int): MimeType? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(name: String): MimeType? = asDynamic()[name]

/**
 * Exposes the JavaScript
[Plugin](https://developer.mozilla.org/en/docs/Web/API/Plugin) to Kotlin
 */
public external abstract class Plugin : ItemArrayLike<MimeType> {
    open val name: String
    open val description: String
    open val filename: String
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
}

@kotlin.internal.InlineOnly
public inline operator fun Plugin.get(index: Int): MimeType? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Plugin.get(name: String): MimeType? = asDynamic()[name]
fun close() { ImageOrientation? /* = ImageOrientation.NONE */ get() = definedExternally
    set(value) = definedExternally
    get() = definedExternally
    var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */ get() = definedExternally
    var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */ get() = definedExternally
    var resizeWidth: Int? get() = definedExternally
    var resizeHeight: Int? get() = definedExternally
    var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */ get() = definedExternally

        val o = js("({})")
        o["imageOrientation"] = imageOrientation
        o["premultiplyAlpha"] = premultiplyAlpha
        o["colorSpaceConversion"] = colorSpaceConversion
        o["resizeWidth"] = resizeWidth
        o["resizeHeight"] = resizeHeight
        return o
    }

    Companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {
        open val data: Any?
        open val origin: String
        open val lastEventId: String
        open val source: UnionMessagePortOrWindowProxy?
        open val ports: Array<MessagePort>

        fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)

        Companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }

        interface MessageEventInit : EventInit {
            var data: Any? /* = null */
                get() = definedExternally
                set(value) = definedExternally
            var origin: String? /* = "" */
                get() = definedExternally
                set(value) = definedExternally
            var lastEventId: String? /* = "" */
                get() = definedExternally
                set(value) = definedExternally
            var source: UnionMessagePortOrWindowProxy? /* = null */
                get() = definedExternally
                set(value) = definedExternally
            var ports: Array<MessagePort>? /* = arrayOf() */
                get() = definedExternally
                set(value) = definedExternally
        }

        @kotlin.internal.InlineOnly
            val o = js("({})")
            o["data"] = data
            o["origin"] = origin
            o["lastEventId"] = lastEventId
            o["source"] = source
            o["ports"] = ports
            o["bubbles"] = bubbles
            o["cancelable"] = cancelable
            o["composed"] = composed
            return o
        }

        /**
         * Exposes the JavaScript [EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin
         */
        public external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
            open val url: String
            open val readyState: Short
            open val bufferedAmount: Number
            var onopen: ((Event) -> dynamic)?
            var onerror: ((Event) -> dynamic)?
            var onclose: ((Event) -> dynamic)?
            fun close() { EventTarget { open
                url: String
                onopen: ((Event) -> dynamic)?
                onerror: ((Event) -> dynamic)?
                onclose: ((Event) -> dynamic)?
            }
        }

        public external interface EventSourceInit {
            var withCredentials: Boolean? /* = false */
                get() = definedExternally
                set(value) = definedExternally
        }

        @kotlin.internal.InlineOnly
        public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
            val o = js("({})")
            o["withCredentials"] = withCredentials
            return o
        }

        /**
         * Exposes the JavaScript [WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin
         */
        public external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
            open val url: String
            open val readyState: Short
            open val bufferedAmount: Number
            var onopen: ((Event) -> dynamic)?
            var onerror: ((Event) -> dynamic)?
            fun close() { EventTarget { open
                url: String
                onopen: ((Event) -> dynamic)?
                onerror: ((Event) -> dynamic)?
            }
        }

        public external interface WebSocketInit {
            var protocols: dynamic? /* = definedExternally */
                get() = definedExternally
                set(value) = definedExternally
        }

        @kotlin.internal.InlineOnly
        public inline fun WebSocketInit(protocols: dynamic? = definedExternally): WebSocketInit {
            val o = js("({})")
            o["protocols"] = protocols
            return o
        }
    }

    Open Source Used In DNAC Platform Arica 9902
String
  var onmessage: ((MessageEvent) -> dynamic)?
fun close(code: Short
= definedExternally, reason: String = definedExternally)
fun send(data: String?)
fun send(data: ArrayBuffer?)
fun send(data: ArrayBufferView?)
fun companion object {
  val CONNECTING: Short
  val OPEN: Short
  val CLOSING: Short
  val CLOSED: Short
}

/**
 * Exposes the
 * [CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin
 */
public external
  open class CloseEvent(type: String?, eventInitDict: CloseEventInit = definedExternally) : Event {
    open val wasClean: Boolean
    open val code: Short
    open val reason: String

    companion object {
      val NONE: Short
      val CAPTURING_PHASE: Short
      val AT_TARGET: Short
      val BUBBLING_PHASE: Short
    }
}

public external interface CloseEventInit : EventInit {
  var wasClean: Boolean? /* = false */
  var code: Short? /* = 0 */
  var reason: String? /* = "" */
}

@kotlin.internal.InlineOnly
public inline fun CloseEventInit(wasClean: Boolean? = false, code: Short? = 0, reason: String? = "", bubbles: Boolean? = false,
cancelable: Boolean? = false): CloseEventInit {
  val o = js("({})")
  o["wasClean"] = wasClean
  o["code"] = code
  o["reason"] = reason
  o["bubbles"] = bubbles
  o["cancelable"] = cancelable
  return o
}

/**
 * Exposes the
 * [MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin
 */
public external
  open class MessageChannel {
    open val port1: MessagePort
    open val port2: MessagePort
}

/**
 * Exposes the
 * [MessagePort](https://developer.mozilla.org/en/docs/Web/API/MessagePort) to Kotlin
 */
public external
  abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy,
  UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()
}

/**
 * Exposes the
 * [BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin
 */
public external
  open class BroadcastChannel(name: String) : EventTarget {
    open val name: String
    var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?)
    fun close()
}

/**
 * Exposes the
 * [WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin
 */
public external
  abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope,
  GlobalPerformance {
    open val self: WorkerGlobalScope
    open val location: WorkerLocation
    open var navigator: WorkerNavigator
    open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    open var onlanguagechange: ((Event) -> dynamic)?
    open var onoffline: ((Event) -> dynamic)?
    open var ononline: ((Event) -> dynamic)?
    open var onrejectionhandled: ((Event) -> dynamic)?
    open var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
    fun importScripts(vararg urls: String?)
}

/**
 * Exposes the
 * [AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin
 */
public external
  interface AbstractWorker {
    var onerror: ((Event) -> dynamic)?
    fun get(): definedExternally
    fun set(value) = definedExternally
}

/**
 * Exposes the
 * [Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin
 */
public external
  abstract class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    var onmessage: ((MessageEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    fun terminate()


String
open var codeBase: String
open var height: String
open var hspace: Int
open var name:
open var object: String
open var vspace: Int
open var width: String

val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript
[HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin
*/
public external abstract class HTMLMarqueeElement : HTMLElement {
    open var behavior: String
    open var bgColor: String
    open var direction: String
    open var height: String
    open var hspace: Int
    open var loop: Int
    open var scrollAmount: Int
    open var scrollDelay: Int
    open var trueSpeed: Boolean
    open var vspace: Int
    open var width: String
    open var onbounce: ((Event) -> dynamic)?
    open var onfinish: ((Event) -> dynamic)?
    open var onstart: ((Event) -> dynamic)?
    fun start()
    fun stop()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin
*/
public external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {
    open var cols: String
    open var rows: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLFrameElement : HTMLElement {
    open var name: String
    open var scrolling: String
    open var src: String
    open var frameBorder: String
    open var longDesc: String
    open var noResize: Boolean
    open val contentDocument: Document?
    open val contentWindow: Window?
    open var marginHeight: String
    open var marginWidth: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin
 */
public external abstract class HTMLFontElement : HTMLElement {
    open var color: String
    open var face: String
open var size: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface External {
    fun AddSearchProvider()
    fun IsSearchProviderInstalled()
}

public external interface EventInit {
var bubbles: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
var cancelable: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
var composed: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun EventInit(bubbles: Boolean? = false,
cancelable: Boolean? = false, composed: Boolean? = false): EventInit {
    val o = js("({})")
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [CustomEvent](https://developer.mozilla.org/en/docs/Web/API/CustomEvent) to Kotlin
 */
public external open class CustomEvent(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {
open val detail: Any?
    fun initCustomEvent(type: String, bubbles: Boolean, cancelable: Boolean, detail: Any?)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CustomEventInit : EventInit {
var detail: Any? /* = null */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CustomEventInit(detail: Any? = null,
    val o = js("({})")
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external interface EventListenerOptions {
var capture: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
var once: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun EventListenerOptions(capture: Boolean? = false,
    once: Boolean? = false): EventListenerOptions {
    val o = js("({})")
    o["capture"] = capture
    o["once"] = once
    return o
}
AddEventListenerOptions {
    val o = js("({})")
    o["passive"] = passive
    o["once"] = once
    o["capture"] = capture
    return o
}

private fun getElementById(elementId: String): Element? =
definedExternally

public external interface DocumentOrShadowRoot {
    fun fullscreenElement: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript
[DocumentOrShadowRoot](https://developer.mozilla.org/en/docs/Web/API/DocumentOrShadowRoot) to Kotlin
*/
public external interface ParentNode {
    fun children: HTMLCollection
    fun firstElementChild: Element?
    fun lastElementChild: Element?
    fun childElementCount: Int
    fun prepend(vararg nodes: dynamic): void
    fun append(vararg nodes: dynamic): void
    fun querySelector(selectors: String): Element?
    fun querySelectorAll(selectors: String): NodeList
}

/**
 * Exposes the JavaScript
[ParentNode](https://developer.mozilla.org/en/docs/Web/API/ParentNode) to Kotlin
*/
public external interface DocumentOrShadowRoot {
    fun valfullscreenElement: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript
[NonElementParentNode](https://developer.mozilla.org/en/docs/Web/API/NonElementParentNode) to Kotlin
*/
public external interface NonElementParentNode {
    fun
getElementById(elementId: String): Element?
}

/**
 * Exposes the JavaScript
[NonDocumentTypeChildNode](https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode) to Kotlin
*/
public external interface NonDocumentTypeChildNode {
    fun
previousElementSibling: Element?
    nextElementSibling: Element?
}

/**
 * Exposes the JavaScript
[ChildNode](https://developer.mozilla.org/en/docs/Web/API/ChildNode) to Kotlin
*/
public external interface ChildNode {
    fun before(vararg nodes: dynamic): void
    fun after(vararg nodes: dynamic): void
    fun replaceWith(vararg nodes: dynamic): void
    fun remove(): void
}

/**
 * Exposes the JavaScript
[Slotable](https://developer.mozilla.org/en/docs/Web/API/Slotable) to Kotlin
*/
public external interface Slotable {
    fun
assignedSlot: HTMLSlotElement?
}

/**
 * Exposes the JavaScript
[NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
*/
public abstract class NodeList : ItemArrayLike<Node> {
    override fun item(index: Int): Node?
}

@kotlin.internal.InlineOnly
public inline operator fun NodeList.get(index: Int): Node? =
asDynamic()[index]

/**
 * Exposes the JavaScript
[HTMLCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLCollection) to Kotlin
*/
public abstract class HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {
    override fun item(index: Int): Element?
    fun namedItem(name: String): Element?
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(index: Int): Element? =
asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(name: String): Element? =
asDynamic()[name]

/**
 * Exposes the JavaScript
[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin
*/
public open class MutationObserver(callback: (Array<MutationRecord>, MutationObserver) -> Unit) {
    fun observe(target: Node, options: MutationObserverInit = definedExternally)
    fun disconnect(): Unit
    fun takeRecords(): Array<MutationRecord>
}

/**
 * Exposes the JavaScript
[MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
*/
public external interface MutationObserverInit {
    var childList: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtree: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    o["subtree"] = subtree
    o["attributeOldValue"] = attributeOldValue
    o["characterDataOldValue"] =
characterDataOldValue o["attributeFilter"] = attributeFilter return o
} /* Exposes the JavaScript [MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin */

public abstract class MutationRecord {
    open val type: String
    open val target: Node
    open val addedNodes: NodeList
    open val removedNodes: NodeList
    open val previousSibling: Node?
    open val nextSibling: Node?
    open val attributeName: String?
    open val attributeNamespace: String?
    open val oldValue: String?
}

/**
 * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin */
public external abstract class Node : EventTarget {
    open val nodeType: Short
    open val nodeName: String
    open val baseURI: String
    open val isConnected: Boolean
    open val ownerDocument: Document
    open val parentNode: Node?
    open val parentElement: Element?
    open val childNodes: NodeList
    open val firstChild: Node?
    open val lastChild: Node?
    open val previousSibling: Node?
    open val nextSibling: Node?
    open var nodeValue: String?
    open var textContent: String?
    fun getRootNode(options: GetRootNodeOptions = definedExternally): Node
    fun hasChildNodes(): Boolean
    fun normalize()
    fun cloneNode(deep: Boolean = definedExternally): Node
    fun isEqualNode(otherNode: Node?): Boolean
    fun isSameNode(otherNode: Node?): Boolean
    fun compareDocumentPosition(other: Node): Short
    fun contains(other: Node?): Boolean
    fun lookupPrefix(namespace: String?): String?
    fun lookupNamespaceURI(prefix: String?): String?
    fun isDefaultNamespace(namespace: String?): Boolean
    fun insertBefore(node: Node, child: Node?): Node
    fun appendChild(node: Node): Node
    fun replaceChild(node: Node, child: Node): Node
    fun removeChild(child: Node): Node

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface GetRootNodeOptions {
    var composed: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean? = false): GetRootNodeOptions {
    val o = js("({})")
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to Kotlin */
    open val implementation: DOMImplementation
    open val URL: String
    open val documentURI: String
    open val origin: String
    open val compatMode: String
    open val characterSet: String
    open val inputEncoding: String
    open val contentType: String
    open val doctype: DocumentType
    open val documentElement: Element
    open val location: Location?
    var domain: String
    var referrer: String
    var cookie: String
    open val lastModified: String
    var title: String
    var dir: String
    var body: HTMLElement?
    open val head: HTMLHeadElement?
    open val images: HTMLCollection
    open val URL: String
    open val documentURI: String
    open val origin: String
    open val compatMode: String
    open val characterSet: String
    open val inputEncoding: String
    open val contentType: String
    open val doctype: DocumentType
    open val documentElement: Element
    open val location: Location?
    var domain: String
    var referrer: String
    var cookie: String
    var lastModified: String
    var title: String
    var dir: String
    var body: HTMLElement
    open val open: Boolean
    open val fullscrenEnabled: Boolean
    open val fullscren: Boolean
    open val onfullscreenchange: ((Event) -> dynamic)
    open val onfullscreenerror: ((Event) -> dynamic)
    override var

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external abstract class DOMImplementation {
    fun createDocumentType(qualifiedName: String?,
        namespace: String?): DocumentType
    fun createDocument(title: String?, doctype: DocumentType?,
        xmlns: String? = definedExternally): XMLDocument
    fun createHTMLDocument(title: String? = definedExternally): Document
    fun hasFeature(): Boolean
}

/**
 * Exposes the JavaScript [DOMImplementation](https://developer.mozilla.org/en/docs/Web/API/DOMImplementation) to Kotlin
 */
public external abstract class DocumentType : Node, ChildNode {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
}

/**
 * Exposes the JavaScript [DocumentFragment](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) to Kotlin
 */
public external abstract class DocumentFragment : Node, NonElementParentNode, ParentNode {
    override val children: HTMLCollection
    override val firstElementChild: Element?
    override val lastElementChild: Element?
    override val childElementCount: Int
    override fun getElementById(elementId: String): Element?
    override fun prepend(vararg nodes: dynamic)
    override fun append(vararg nodes: dynamic)
    override fun querySelector(selectors: String): Element?
    override fun querySelectorAll(selectors: String): NodeList
}

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
}

/**
 * Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin
 */
public external abstract class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {
    val mode: ShadowRootMode
    override val fullscreenElement: Element?
}

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
}
[Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin
*/
public external abstract class

public external interface 

@kotlin.internal.InlineOnly
public inline fun ShadowRootInit(mode: ShadowRootMode?): ShadowRootInit {
    val o = js("({})")
    o["mode"] = mode
    return o
}

/**
 * Exposes the JavaScript [NamedNodeMap](https://developer.mozilla.org/en/docs/Web/API/NamedNodeMap) to Kotlin
 */
public external abstract class NamedNodeMap : ItemArrayLike<Attr> {
    fun getNamedItemNS(namespace: String?, localName: String): Attr?
    fun setNamedItem(attr: Attr): Attr?
    fun setNamedItemNS(attr: Attr): Attr?
    fun removeNamedItem(qualifiedName: String): Attr
    fun removeNamedItemNS(namespace: String?, localName: String): Attr
    override fun item(index: Int): Attr?
    fun getNamedItem(qualifiedName: String): Attr?
}

@kotlin.internal.InlineOnly
public inline operator fun
NamedNodeMap.get(index: Int): Attr? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator

fun NamedNodeMap.get(qualifiedName: String): Attr? = asDynamic()[qualifiedName]

/**
 * Exposes the JavaScript [Attr](https://developer.mozilla.org/en/docs/Web/API/Attr) to Kotlin
 */
public external abstract class Attr : Node {
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val name: String
    open var value: String
    open val ownerElement: Element?
    open val specified: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [CharacterData](https://developer.mozilla.org/en/docs/Web/API/CharacterData) to Kotlin
 */
public external abstract class CharacterData : Node, NonDocumentTypeChildNode, ChildNode {
    open var data: String
    open val length: Int
    fun substringData(offset: Int, count: Int): String
    fun appendData(data: String)
    fun insertData(offset: Int, data: String)
    fun deleteData(offset: Int, count: Int)
    fun replaceData(offset: Int, count: Int, data: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin
 */
public external open class Text(data: String = definedExternally) : CharacterData, Slotable, GeometryUtils {
    open val wholeText: String
    override val assignedSlot: HTMLSlotElement?
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    fun splitText(offset: Int): Text
    override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */) : Array<DOMQuad>
    override fun convertQuadFromNode(quad: DOMQuad, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */) : DOMPoint
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
public external open class CDATASection : Text {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {
    open val target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external open class Comment(data: String = definedExternally) : CharacterData {
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external open class Range {
    open val startContainer: Node
    open val startOffset: Int
    open val endContainer: Node
    open val endOffset: Int
    open val collapsed: Boolean
    open val commonAncestorContainer: Node
    fun setStart(node: Node, offset: Int)
    fun setEnd(node: Node, offset: Int)
    fun setStartBefore(node: Node)
    fun setStartAfter(node: Node)
    fun setEndBefore(node: Node)
    fun setEndAfter(node: Node)
    fun collapse(toStart: Boolean = definedExternally)
    fun selectNode(node: Node)
    fun selectNodeContents(node: Node)
    fun compareBoundaryPoints(how: Short, sourceRange: Range): Short
    fun deleteContents()
    fun extractContents(): DocumentFragment
    fun cloneContents(): DocumentFragment
    fun insertNode(node: Node)
    fun surroundContents(newParent: Node)
    fun isPointInRange(node: Node, offset: Int): Boolean
    fun comparePoint(node: Node, offset: Int): Short
    fun intersectsNode(node: Node): Boolean
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun createContextualFragment(fragment: String): DocumentFragment

[CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin

### Open Source Used In DNAC Platform Arica 9914

[ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin

[Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin

[Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
companion object {
    val START_TO_START: Short
    val START_TO_END: Short
    val END_TO_END: Short
    val END_TO_START: Short
}

/**
 * Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
    open val root: Node
    open val referenceNode: Node
    open val pointerBeforeReferenceNode: Boolean
    open val whatToShow: Int
    open val filter: NodeFilter?
    fun nextNode(): Node?
    fun previousNode(): Node?
    fun detach()
}

/**
 * Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
    open val root: Node
    open val whatToShow: Int
    open val filter: NodeFilter?
    open var currentNode: Node
    fun parentNode(): Node?
    fun firstChild(): Node?
    fun lastChild(): Node?
    fun previousSibling(): Node?
    fun nextSibling(): Node?
    fun previousNode(): Node?
    fun nextNode(): Node?
}

/**
 * Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
    fun acceptNode(node: Node): Short

    companion object {
        val FILTER_ACCEPT: Short
        val FILTER_REJECT: Short
        val FILTER_SKIP: Short
        val SHOW_ALL: Int
        val SHOW_ELEMENT: Int
        val SHOW_ATTRIBUTE: Int
        val SHOW_TEXT: Int
        val SHOW_CDATA_SECTION: Int
        val SHOW_ENTITY_REFERENCE: Int
        val SHOW_ENTITY: Int
        val SHOW_PROCESSING_INSTRUCTION: Int
        val SHOW_COMMENT: Int
        val SHOW_DOCUMENT: Int
        val SHOW_DOCUMENT_TYPE: Int
        val SHOW_DOCUMENT_FRAGMENT: Int
        val SHOW_NOTATION: Int
    }
}

/**
 * Exposes the JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
 */
public external abstract class DOMTokenList : ItemArrayLike<String> {
    open var value: String
    fun contains(token: String): Boolean
    fun add(vararg tokens: String)
    fun remove(vararg tokens: String)
    fun toggle(token: String, force: Boolean = definedExternally): Boolean
    fun replace(token: String, newToken: String)
    fun supports(token: String): Boolean
    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript [DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin
 */
public external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {
    override var x: Double
    override var y: Double
    override var width: Double
    override var height: Double
}

@kotlin.internal.InlineOnly
public inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["z"] = z
    o["w"] = w
    return o
}

// Other code elements are not shown for brevity.
Open Source Used In DNAC Platform Arica 9916

[DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin

```
public external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {
    open val x: Double
    open val y: Double
    open val width: Double
    open val height: Double
}
```

```
@kotlin.internal.InlineOnly
public inline fun DOMRectInit(x: Double? = 0.0, y: Double? = 0.0, width: Double? = 0.0, height: Double? = 0.0): DOMRectInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["width"] = width
    o["height"] = height
    return o
}
```

```
public external interface DOMRectList : ItemArrayLike<DOMRect> {
    override fun item(index: Int): DOMRect?
}
```

```
@kotlin.internal.InlineOnly
public inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]
```

```
/**
 * Exposes the JavaScript [DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin
 */
public external open class DOMQuad {
    constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)
    constructor(rect: DOMRectInit)
    open val p1: DOMPoint
    open val p2: DOMPoint
    open val p3: DOMPoint
    open val p4: DOMPoint
    open val bounds: DOMRectReadOnly
}
```

```
/**
 * Exposes the JavaScript [DOMMatrixReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMMatrixReadOnly) to Kotlin
 */
public external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {
    open val a: Double
    open val b: Double
    open val c: Double
    open val d: Double
    open val e: Double
    open val f: Double
    open val m11: Double
    open val m12: Double
    open val m13: Double
    open val m14: Double
    open val m21: Double
    open val m22: Double
    open val m23: Double
    open val m24: Double
    open val m31: Double
    open val m32: Double
    open val m33: Double
    open val m34: Double
    open val m41: Double
    open val m42: Double
    open val m43: Double
    open val m44: Double
    open val is2D: Boolean
    open val isIdentity: Boolean
    fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun rotate(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVector(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngle(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
    fun skewX(sx: Double): DOMMatrix
    fun skewY(sy: Double): DOMMatrix
    fun multiply(other: DOMMatrix): DOMMatrix
    fun multiplySelf(other: DOMMatrix): DOMMatrix
    fun inverse(): DOMMatrix
    fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint
    fun toFloat32Array(): Float32Array
    fun toFloat64Array(): Float64Array
}
```

```
/**
 * Exposes the JavaScript [DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin
 */
public external class DOMMatrix() : DOMMatrixReadOnly {
    constructor(transformList: String)
    constructor(other: DOMMatrixReadOnly)
    constructor(array32: Float32Array)
    constructor(array64: Float64Array)
    constructor(numberSequence: Array<Double>)
    fun preMultiplySelf(other: DOMMatrix): DOMMatrix
}```
DOMMatrix
  fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
  fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
  fun scale3dSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
  fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
  fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
  fun rotateFromVectorSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
  fun skewXSelf(sx: Double): DOMMatrix
  fun skewYSelf(sy: Double): DOMMatrix
  fun invertSelf(): DOMMatrix
  fun setMatrixValue(transformList: String): DOMMatrix
}

public external interface ScrollOptions {
    var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {
    var o = js("({})")
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [ScrollToOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollToOptions) to Kotlin
 */
public external interface ScrollToOptions : ScrollOptions {
    var left: Double?
    get() = definedExternally
    set(value) = definedExternally
    var top: Double?
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    var o = js("({})")
    o["left"] = left
    o["top"] = top
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
    open val media: String
    open val matches: Boolean
    open var onchange: ((Event) -> dynamic)?

    fun addListener(listener: EventListener?)
    fun addListener(listener: ((Event) -> Unit)?)
    fun removeListener(listener: EventListener?)
    fun removeListener(listener: ((Event) -> Unit)?)
}

/**
 * Exposes the JavaScript [MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin
 */
public external open class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event {
    open val media: String
    open val matches: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaQueryListEventInit : EventInit {
    var media: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
    var matches: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    var o = js("({})")
    o["media"] = media
    o["matches"] = matches
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin
 */
public external abstract class Screen {
    open val availWidth: Int
    open val availHeight: Int
    open val width: Int
    open val height: Int
    open val colorDepth: Int
    open val pixelDepth: Int
}

/**
 * Exposes the JavaScript [CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin
 */
public external abstract class CaretPosition {
    open val offsetNode: Node
    open val offset: Int
    fun getClientRect(): DOMRect?
}

public external interface ScrollIntoViewOptions : ScrollOptions {
    var block: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
    get() = definedExternally
    set(value) = definedExternally
    var inline: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    var o = js("({})")
    o["block"] = block
    o["inline"] = inline
    return o
}
((MouseEvent) -> dynamic)?
override var onmouseenter: ((MouseEvent) -> dynamic)?
override var onmouseleave: ((MouseEvent) -> dynamic)?
override var onmousemove: ((MouseEvent) -> dynamic)?
override var onmouseout: ((MouseEvent) -> dynamic)?
override var onmouseover: ((MouseEvent) -> dynamic)?
override var onmouseup: ((MouseEvent) -> dynamic)?
override var onwheel: ((WheelEvent) -> dynamic)?
override var onpause: ((Event) -> dynamic)?
override var onplay: ((Event) -> dynamic)?
override var onratechange: ((Event) -> dynamic)?
override var onreset: ((Event) -> dynamic)?
override var onresize: ((Event) -> dynamic)?
override var onscroll: ((Event) -> dynamic)?
override var onseeked: ((Event) -> dynamic)?
override var onseeking: ((Event) -> dynamic)?
override var onsubmit: ((Event) -> dynamic)?
override var onsuspend: ((Event) -> dynamic)?
override var onontoggle: ((Event) -> dynamic)?
override var onvolumechange: ((Event) -> dynamic)?
override var onwaiting: ((Event) -> dynamic)?
override var oncontextmenu: ((Event) -> dynamic)?
override var onkeydown: ((KeyboardEvent) -> dynamic)?
override var onkeypress: ((KeyboardEvent) -> dynamic)?
override var onkeyup: ((KeyboardEvent) -> dynamic)?
override var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
override var onkeydown: ((KeyboardEvent) -> dynamic)?
override var onkeypress: ((KeyboardEvent) -> dynamic)?
override var onkeyup: ((KeyboardEvent) -> dynamic)?
override var onload: ((Event) -> dynamic)?
override var onunload: ((Event) -> dynamic)?

override var onpointerdown: ((PointerEvent) -> dynamic)?
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointermove: ((PointerEvent) -> dynamic)?
override var onpointerenter: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncontextmenu: ((Event) -> dynamic)?
override var oncopy: ((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override var contentEditable: String
override val isContentEditable: Boolean
override val style: CSSStyleDeclaration
override val children: HTMLCollection
override val firstElementChild: Element?
override val lastElementChild: Element?
override val childElementCount: Int
override val previousElementSibling: Element?
override val nextElementSibling: Element?
override val assignedSlot: HTMLSlotElement?
override fun prepend(vararg nodes: dynamic)
override fun append(vararg nodes: dynamic)
override fun querySelector(selectors: String): Element?
override fun querySelectorAll(selectors: String): NodeList
override fun before(vararg nodes: dynamic)
override fun after(vararg nodes: dynamic)
override fun replaceWith(vararg nodes: dynamic)
override fun moveAfter(vararg nodes: dynamic)
override fun moveBefore(vararg nodes: dynamic)
override fun replace(vararg nodes: dynamic)
override fun remove(): Unit
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val ENTITY_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external open class Audio(src: String = definedExternally): HTMLAudioElement

Audio(src: String = definedExternally): HTMLAudioElement
Open Source Used In DNAC Platform Arica 9923

public inline val TextTrackMode.Companion.DISABLED: TextTrackMode get() = "disabled".asDynamic().unsafeCast<TextTrackMode>()

public inline val TextTrackMode.Companion.HIDDEN: TextTrackMode get() = "hidden".asDynamic().unsafeCast<TextTrackMode>()

public inline val TextTrackMode.Companion.SHOWING: TextTrackMode get() = "showing".asDynamic().unsafeCast<TextTrackMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface TextTrackKind {
    companion object
}

public inline val TextTrackKind.Companion.SUBTITLES: TextTrackKind get() = "subtitles".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.CAPTIONS: TextTrackKind get() = "captions".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.DESCRIPTIONS: TextTrackKind get() = "descriptions".asDynamic().unsafeCast<TextTrackKind>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SelectionMode {
    companion object
}

public inline val SelectionMode.Companion.SELECT: SelectionMode get() = "select".asDynamic().unsafeCast<SelectionMode>()

public inline val SelectionMode.Companion.START: SelectionMode get() = "start".asDynamic().unsafeCast<SelectionMode>()

public inline val SelectionMode.Companion.END: SelectionMode get() = "end".asDynamic().unsafeCast<SelectionMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasFillRule {
    companion object
}

public inline val CanvasFillRule.Companion.NONZERO: CanvasFillRule get() = "nonzero".asDynamic().unsafeCast<CanvasFillRule>()

public inline val CanvasFillRule.Companion.EVENODD: CanvasFillRule get() = "evenodd".asDynamic().unsafeCast<CanvasFillRule>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ImageSmoothingQuality {
    companion object
}


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineCap {
    companion object
}

public inline val CanvasLineCap.Companion.BUTT: CanvasLineCap get() = "butt".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.ROUND: CanvasLineCap get() = "round".asDynamic().unsafeCast<CanvasLineCap>()

public inline val CanvasLineCap.Companion.SQUARE: CanvasLineCap get() = "square".asDynamic().unsafeCast<CanvasLineCap>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineJoin {
    companion object
}

public inline val CanvasLineJoin.Companion.BEVEL: CanvasLineJoin get() = "bevel".asDynamic().unsafeCast<CanvasLineJoin>()

public inline val CanvasLineJoin.Companion.MITER: CanvasLineJoin get() = "miter".asDynamic().unsafeCast<CanvasLineJoin>()
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResizeQuality {
    companion object
}

public inline val ResizeQuality.Companion.PIXELATED: ResizeQuality get() = "pixelated".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.LOW: ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.MEDIUM: ResizeQuality get() = "medium".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.HIGH: ResizeQuality get() = "high".asDynamic().unsafeCast<ResizeQuality>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface BinaryType {
    companion object
}

public inline val BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()

public inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() = "arraybuffer".asDynamic().unsafeCast<BinaryType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WorkerType {
    companion object
}

public inline val WorkerType.Companion.CLASSIC: WorkerType get() = "classic".asDynamic().unsafeCast<WorkerType>()

public inline val WorkerType.Companion.MODULE: WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ShadowRootMode {
    companion object
}

public inline val ShadowRootMode.Companion.OPEN: ShadowRootMode get() = "open".asDynamic().unsafeCast<ShadowRootMode>()

public inline val ShadowRootMode.Companion.CLOSED: ShadowRootMode get() = "closed".asDynamic().unsafeCast<ShadowRootMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollBehavior {
    companion object
}

public inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() = "auto".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() = "instant".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() = "smooth".asDynamic().unsafeCast<ScrollBehavior>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollLogicalPosition {
    companion object
}


public inline val ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() = "center".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() = "end".asDynamic().unsafeCast<ScrollLogicalPosition>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CSSBoxType {
    companion object
}

public inline val CSSBoxType.Companion.MARGIN: CSSBoxType get() = "margin".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.BORDER: CSSBoxType get() = "border".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.CONTENT: CSSBoxType get() = "content".asDynamic().unsafeCast<CSSBoxType>()
```kotlin
/**
 * Exposes the JavaScript Headers to Kotlin
 */
public external open class Headers(init: dynamic = definedExternally) {
    fun append(name: String, value: String)
    fun delete(name: String)
    fun get(name: String): String?
    fun has(name: String): Boolean
    fun set(name: String, value: String)
}

/**
 * Exposes the JavaScript Body to Kotlin
 */
public external interface Body {
    val bodyUsed: Boolean
    fun arrayBuffer(): Promise<ArrayBuffer>
    fun blob(): Promise<Blob>
    fun formData(): Promise<FormData>
    fun json(): Promise<Any?>
    fun text(): Promise<String>
}

/**
 * Exposes the JavaScript Request to Kotlin
 */
public external open class Request(input: dynamic, init: RequestInit = definedExternally) : Body {
    open val method: String
    open val url: String
    open val headers: Headers
    open val type: RequestType
    open val destination: RequestDestination
    open val referrer: String
    open val referrerPolicy: dynamic
    open val mode: RequestMode
    open val credentials: RequestCredentials
    open val cache: RequestCache
    open val redirect: RequestRedirect
    open val integrity: String
    open val keepalive: Boolean
    override val bodyUsed: Boolean
    fun clone(): Request
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun formData(): Promise<FormData>
    override fun json(): Promise<Any?>
    override fun text(): Promise<String>
}

public external interface RequestInit {
    var method: String?
    var headers: dynamic
    var body: dynamic
    var referrer: String?
    var referrerPolicy: dynamic
    var mode: RequestMode?
    var credentials: RequestCredentials?
    var cache: RequestCache?
    var redirect: RequestRedirect?
    var integrity: String?
    var keepalive: Boolean?
    var window: Any?
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["method"] = method
    o["headers"] = headers
    o["body"] = body
    o["referrer"] = referrer
    o["referrerPolicy"] = referrerPolicy
    o["mode"] = mode
    o["credentials"] = credentials
    o["cache"] = cache
    o["redirect"] = redirect
    o["integrity"] = integrity
    o["keepalive"] = keepalive
    o["window"] = window
    return o
}

/**
 * Exposes the JavaScript Response to Kotlin
 */
public external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body {
    open val type: ResponseType
    open val url: String
    open val status: Short
    open val ok: Boolean
    open val statusText: String
    open val headers: Headers
    open val body: dynamic
    open val trailer: Promise<Headers>
    open valło: Promise<Headers>
    fun clone(): Response
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun formData(): Promise<FormData>
    override fun json(): Promise<Any?>
    override fun text(): Promise<String>

    companion object {
        fun error(): Response
        fun redirect(url: String, status: Short = definedExternally): Response
    }
}
```
RequestDestination.Companion.XSLT: RequestDestination get() = "xslt".asDynamic().unsafeCast&lt;RequestDestination&gt;()/* please, don't implement this interface! */

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestMode {

    companion object
}

public inline val RequestMode.Companion.NAVIGATE: RequestMode get() = "navigate".asDynamic().unsafeCast&lt;RequestMode&gt;()/* please, don't implement this interface! */

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCredentials {
    companion object
}

public inline val RequestCredentials.Companion.OMIT: RequestCredentials get() = "omit".asDynamic().unsafeCast&lt;RequestCredentials&gt;()/* please, don't implement this interface! */

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCache {
    companion object
}

public inline val RequestCache.Companion.DEFAULT: RequestCache get() = "default".asDynamic().unsafeCast&lt;RequestCache&gt;()/* please, don't implement this interface! */

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestRedirect {
    companion object
}

public inline val RequestRedirect.Companion.FOLLOW: RequestRedirect get() = "follow".asDynamic().unsafeCast&lt;RequestRedirect&gt;()/* please, don't implement this interface! */

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResponseType {
    companion object
}

public inline val ResponseType.Companion.BASIC: ResponseType get() = "basic".asDynamic().unsafeCast&lt;ResponseType&gt;()/* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

Open Source Used In DNAC Platform Arica 9928
kotlin.js/*unimport org.khronos.webgl.*unimport org.w3c.css.masking.*unimport org.w3c.dom.*unimport org.w3c.dom.clipboard.*unimport org.w3c.dom.css.*unimport org.w3c.dom.encryptedmedia.*unimport org.w3c.dom.events.*unimport org.w3c.dom.mediasearch.*unimport org.w3c.dom.parsing.*unimport org.w3c.dom.pointerevents.*unimport org.w3c.dom.svg.*unimport org.w3c.dom.url.*unimport org.w3c.fetch.*unimport org.w3c.files.*unimport org.w3c.notifications.*unimport org.w3c.performance.*unimport org.w3c.workers.*unimport org.w3c.xhr.**/un/** * Exposes the JavaScript

[MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin

```kotlin
/**
 * Exposes the JavaScript [MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin
 */
public external open class MediaStream() : EventTarget, MediaProvider {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}

/**
 * Exposes the JavaScript [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin
 */
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open var muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open val readyState: MediaStreamTrackState
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop(): Array<MediaStreamTrack>
    fun getCapabilities(): MediaStreamCapabilities
    fun getSettings(): MediaStreamSettings
    fun applyConstraints(constraints: MediaStreamTrackSupportedConstraints = definedExternally): Promise<Unit>
}

/**
 * Exposes the JavaScript [MediaStreamTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackSupportedConstraints) to Kotlin
 */
public external interface MediaStreamTrackSupportedConstraints {
    var width: Boolean? /* = true */
    var height: Boolean? /* = true */
    var aspectRatio: Boolean? /* = true */
    var frameRate: Boolean? /* = true */
    var facingMode: Boolean? /* = true */
    var resizeMode: Boolean? /* = true */
    var volume: Boolean? /* = true */
    var sampleRate: Boolean? /* = true */
    var sampleSize: Boolean? /* = true */
    var echoCancellation: Boolean? /* = true */
    var autoGainControl: Boolean? /* = true */
    var noiseSuppression: Boolean? /* = true */
    var latency: Boolean? /* = true */
    var channelCount: Boolean? /* = true */
    var deviceId: Boolean? /* = true */
    var groupId: Boolean? /* = true */
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    return o
}
```
openSource

```javascript
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

public external interface MediaTrackCapabilities {
    in var width: ULongRange?
    get() = definedExternally
    set(value) = definedExternally
    var height: ULongRange?
    get() = definedExternally
    set(value) = definedExternally
    var aspectRatio: DoubleRange?
    get() = definedExternally
    set(value) = definedExternally
    var frameRate: DoubleRange?
    get() = definedExternally
    set(value) = definedExternally
    var facingMode: Array<String>?
    get() = definedExternally
    set(value) = definedExternally
    var resizeMode: Array<String>?
    get() = definedExternally
    set(value) = definedExternally
    var volume: DoubleRange?
    get() = definedExternally
    set(value) = definedExternally
    var echoCancellation: Array<Boolean>?
    get() = definedExternally
    set(value) = definedExternally
    var autoGainControl: Array<Boolean>?
    get() = definedExternally
    set(value) = definedExternally
    var noiseSuppression: Array<Boolean>?
    get() = definedExternally
    set(value) = definedExternally
    var latency: DoubleRange?
    get() = definedExternally
    set(value) = definedExternally
    var channelCount: ULongRange?
    get() = definedExternally
    set(value) = definedExternally
    var deviceId: String?
    get() = definedExternally
    set(value) = definedExternally
    var groupId: String?
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript
[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin
 */
public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced: Array<MediaTrackConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}
```

Open Source Used In DNAC Platform Arica 9930
definedExternally
var height: dynamic
   get() = definedExternally
   set(value) = definedExternally
var aspectRatio: dynamic
   get() = definedExternally
   set(value) = definedExternally
var frameRate: dynamic
   get() = definedExternally
   set(value) = definedExternally
var facingMode: dynamic
   get() = definedExternally
   set(value) = definedExternally
var resizeMode: dynamic
   get() = definedExternally
   set(value) = definedExternally
var volume: dynamic
   get() = definedExternally
   set(value) = definedExternally
var sampleRate: dynamic
   get() = definedExternally
   set(value) = definedExternally
var resizeMode: dynamic
   get() = definedExternally
   set(value) = definedExternally
var latency: dynamic
   get() = definedExternally
   set(value) = definedExternally
var channelCount: dynamic
   get() = definedExternally
   set(value) = definedExternally
var deviceId: dynamic
   get() = definedExternally
   set(value) = definedExternally
var groupId: dynamic
   get() = definedExternally
   set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript
[MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
       get() = definedExternally
       set(value) = definedExternally
    var height: Int?
       get() = definedExternally
       set(value) = definedExternally
    var aspectRatio: Double?
       get() = definedExternally
       set(value) = definedExternally
    var frameRate: Double?
       get() = definedExternally
       set(value) = definedExternally
    var facingMode: String?
       get() = definedExternally
       set(value) = definedExternally
    var resizeMode: String?
       get() = definedExternally
       set(value) = definedExternally
    var volume: Double?
       get() = definedExternally
       set(value) = definedExternally
    var sampleRate: Int?
       get() = definedExternally
       set(value) = definedExternally
    var sampleSize: Int?
       get() = definedExternally
       set(value) = definedExternally
    var echoCancellation: Boolean?
       get() = definedExternally
       set(value) = definedExternally
    var autoGainControl: Boolean?
       get() = definedExternally
       set(value) = definedExternally
    var noiseSuppression: Boolean?
       get() = definedExternally
       set(value) = definedExternally
    var latency: Double?
       get() = definedExternally
       set(value) = definedExternally
    var channelCount: Int?
       get() = definedExternally
       set(value) = definedExternally
    var deviceId: String?
       get() = definedExternally
       set(value) = definedExternally
    var groupId: String?
       get() = definedExternally
       set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
}
frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript [MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin
 */
public external open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) :
    Event {
        open var track: MediaStreamTrack

        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }
        open val'*'o \npublic external interface MediaStreamTrackEventInit : EventInit {
            var track: MediaStreamTrack?
                get() = definedExternally
                set(value) = definedExternally
        }

        public external abstract class OverconstrainedErrorEvent(type: String, eventInitDict:
            OverconstrainedErrorEventInit) : Event {
                open var error: dynamic

                companion object {
                    val NONE: Short
                    val CAPTURING_PHASE: Short
                    val AT_TARGET: Short
                    val BUBBLING_PHASE: Short
                }

                public external interface OverconstrainedErrorEventInit : EventInit {
                    var error: dynamic /* = null */
                        get() = definedExternally
                        set(value) = definedExternally
                }

                /**
                 * Exposes the JavaScript [MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin
                 */
                public external abstract class MediaDevices : EventTarget {
                    open var ondevicechange: ((Event) -> dynamic)?
                    fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>
                    fun getSupportedConstraints(): MediaTrackSupportedConstraints
                    fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>
                }

                /**
                 * Exposes the JavaScript [MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin
                 */
                public external abstract class MediaDeviceInfo {
                    open val deviceId: String
                    open val kind: MediaDeviceKind

                    open val label: String
                    fun toJSON(): dynamic
                }

                public external class InputDeviceInfo : MediaDeviceInfo {
                    fun getCapabilities(): MediaTrackCapabilities
                }

                /**
                 * Exposes the JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin
                 */
                public external interface MediaStreamConstraints {
                    var video: dynamic /* = false */
                        get() = definedExternally
                        set(value) = definedExternally
                    var audio: dynamic /* = false */
                        get() = definedExternally
                        set(value) = definedExternally
                }

                @kotlin.internal.InlineOnly
                public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
                    val o = js("({})")
                    o["video"] = video
                    o["audio"] = audio
                    return o
                }

                public external interface ConstrainablePattern {
                    var onoverconstrained: ((Event) -> dynamic)?
                        get() = definedExternally
                        set(value) = definedExternally
                    fun getCapabilities(): Capabilities
                    fun getConstraints(): Constraints
                    fun getSettings(): Settings
                    fun applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>
                }

                /**
                 * Exposes the JavaScript [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin
                 */
                public external interface DoubleRange {
                    var max: Double
                        get() = definedExternally
                        set(value) = definedExternally
                    var min: Double
                        get() = definedExternally
                        set(value) = definedExternally

                    @kotlin.internal.InlineOnly
                    public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
                        val o = js("({})")
                        o["max"] = max
                        o["min"] = min
                        return o
                    }
                }
var ideal: Double?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainDoubleRange

ConstrainBooleanParameters

ConstrainBooleanParameters

ConstrainDOMStringParameters

ConstrainDOMStringParameters

Constraints

Constraints

Constraints

MediaStreamTrackState

MediaStreamTrackState

MediaStreamTrackState

MediaStreamTrackState

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

VideoFacingModeEnum

Open Source Used In DNAC Platform Arica 9933
"left".asDynamic().unsafeCast<VideoFacingModeEnum>().

public inline val VideoFacingModeEnum.Companion.RIGHT: VideoFacingModeEnum get() = "right".asDynamic().unsafeCast<VideoFacingModeEnum>().

/* please, don't implement this interface! */


/* please, don't implement this interface! */

/* please, don't implement this interface! */

public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>().

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>().

public inline val MediaDeviceKind.Companion.VIDEOINPUT: MediaDeviceKind get() = "videoinput".asDynamic().unsafeCast<MediaDeviceKind>().

@kotlin.internal.InlineOnly
public inline operator fun SourceBufferList.get(index: Int): SourceBuffer? = asDynamic()[index]

/*@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")*/

/* please, don't implement this interface! */

/* please, don't implement this interface! */

/* please, don't implement this interface! */

/* please, don't implement this interface! */
EndOfStreamError {\n    companion object\n}\n
public inline val EndOfStreamError.Companion.NETWORK:
    EndOfStreamError get() = "network".asDynamic().unsafeCast<EndOfStreamError>()

public inline val
    EndOfStreamError.Companion.DECODE: EndOfStreamError get() = "decode".asDynamic().unsafeCast<EndOfStreamError>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface AppendMode {

    companion object
}

public inline val
    AppendMode.Companion.SEGMENTS: AppendMode get() = "segments".asDynamic().unsafeCast<AppendMode>()

public inline val
    AppendMode.Companion.SEQUENCE: AppendMode get() = "sequence".asDynamic().unsafeCast<AppendMode>()

/**
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.dom.pointerevents

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.gifs.*
import org.w3c.dom.js EventListener
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface PointerEventInit : MouseEventInit {
    var pointerId: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var width: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var height: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var pressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tangentialPressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tiltX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var tiltY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var twist: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var pointerType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isPrimary: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun PointerEventInit(pointerId: Int? = 0, width:
    Double? = 1.0, height: Double? = 1.0, pressure: Float? = 0f, tangentialPressure: Float? = 0f, tiltX: Int? = 0, tiltY:
    Int? = 0, twist: Int? = 0, pointerType: String? = "", isPrimary: Boolean? = false, screenX: Int? = 0, screenY:
    Int? = 0, clientX: Int? = 0, clientY: Int? = 0, button: Short? = 0, buttons: Short? = 0, relatedTarget: EventTarget? = null,
    region: String? = null, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false, metaKey:
    Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false, modifierFn: Boolean? = false,
    modifierFnLock: Boolean? = false, modifierHyper: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock:
    val o = js("({})")
        o["pointerId"] = pointerId
        o["width"] = width
        o["height"] = height
        o["pressure"] = pressure
        o["tangentialPressure"] = tangentialPressure
        o["tiltX"] = tiltX
        o["tiltY"] = tiltY
        o["twist"] = twist
        o["isPrimary"] = isPrimary
        o["screenX"] = screenX
        o["screenY"] = screenY
        o["clientX"] = clientX
        o["clientY"] = clientY
        o["button"] = button
        o["buttons"] = buttons
        o["relatedTarget"] = relatedTarget
        o["region"] = region
        o["ctrlKey"] = ctrlKey
        o["shiftKey"] = shiftKey
        o["altKey"] = altKey
        o["metaKey"] = metaKey
        o["modifierAltGraph"] = modifierAltGraph
        o["modifierCapsLock"] = modifierCapsLock
        o["modifierFn"] = modifierFn
        o["modifierFnLock"] = modifierFnLock
        o["modifierHyper"] = modifierHyper
        o["modifierNumLock"] = modifierNumLock
        o["modifierScrollLock"] = modifierScrollLock
        o["modifierSuper"] = modifierSuper
        o["modifierSymbol"] = modifierSymbol
        o["modifierSymbolLock"] = modifierSymbolLock
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] =
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin
 */
public external abstract class SVGGeometryElement : SVGGraphicsElement {
    open val pathLength: Float
    fun isPointInFill(point: DOMPoint): Boolean
    fun isPointInStroke(point: DOMPoint): Boolean
    fun getTotalLength(): Float
    fun getPointAtLength(distance: Float): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGNumber](https://developer.mozilla.org/en/docs/Web/API/SVGNumber) to Kotlin
 */
public external abstract class SVGNumber {
    open var value: Float
}

/**
 * Exposes the JavaScript [SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin
 */
public external abstract class SVGLength {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_LENGTHTYPE_UNKNOWN: Short
        val SVG_LENGTHTYPE_NUMBER: Short
        val SVG_LENGTHTYPE_PERCENTAGE: Short
        val SVG_LENGTHTYPE_EMS: Short
        val SVG_LENGTHTYPE_EXS: Short
        val SVG_LENGTHTYPE_PX: Short
        val SVG_LENGTHTYPE_CM: Short
        val SVG_LENGTHTYPE_MM: Short
        val SVG_LENGTHTYPE_IN: Short
        val SVG_LENGTHTYPE_PT: Short
        val SVG_LENGTHTYPE_PC: Short
    }
}

/**
 * Exposes the JavaScript [SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin
 */
public external abstract class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_ANGLETYPE_UNKNOWN: Short
        val SVG_ANGLETYPE_UNSPECIFIED: Short
        val SVG_ANGLETYPE_DEG: Short
        val SVG_ANGLETYPE_RAD: Short
        val SVG_ANGLETYPE_GRAD: Short
    }
}

public external abstract class SVGNameList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: dynamic): dynamic
    fun insertItemBefore(newItem: dynamic, index: Int): dynamic
    fun replaceItem(newItem: dynamic, index: Int): dynamic
    fun removeItem(index: Int): dynamic
    fun appendItem(newItem: dynamic): dynamic
    fun getItem(index: Int): dynamic
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin
 */
public external abstract class SVGNumberList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGNumber): SVGNumber
    fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber
    fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber
    fun removeItem(index: Int): SVGNumber
    fun appendItem(newItem: SVGNumber): SVGNumber
    fun getItem(index: Int): SVGNumber
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }
Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin

```kotlin
public external abstract class SVGLengthList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGLength): SVGLength
    fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength
    fun removeItem(index: Int): SVGLength
    fun appendItem(newItem: SVGLength): SVGLength
    fun getItem(index: Int): SVGLength
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList[get](index: Int): SVGLength = asDynamic()[index]
```

Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin

```kotlin
public external abstract class SVGAnimatedBoolean {
    open var baseVal: Boolean
    open val animVal: Boolean
}
```

Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin

```kotlin
public external abstract class SVGAnimatedEnumeration {
    open var baseVal: Short
    open val animVal: Short
}
```

Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin

```kotlin
public external abstract class SVGAnimatedInteger {
    open var baseVal: Int
    open val animVal: Int
}
```

Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin

```kotlin
public external abstract class SVGAnimatedNumber {
    open var baseVal: Float
    open val animVal: Float
}
```

Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin

```kotlin
public external abstract class SVGAnimatedLength {
    open val baseVal: SVGLength
    open val animVal: SVGLength
}
```

Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin

```kotlin
public external abstract class SVGAnimatedAngle {
    open val baseVal: SVGAngle
    open val animVal: SVGAngle
}
```

Exposes the JavaScript [SVGAnimatedString](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedString) to Kotlin

```kotlin
public external abstract class SVGAnimatedString {
    open var baseVal: String
    open val animVal: String
}
```

Exposes the JavaScript [SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect) to Kotlin

```kotlin
public external abstract class SVGAnimatedRect {
    open val baseVal: DOMRect
    open val animVal: DOMRectReadOnly
}
```

Exposes the JavaScript [SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin

```kotlin
public external abstract class SVGAnimatedNumberList {
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberList
}
```

Exposes the JavaScript [SVGAnimatedLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLengthList) to Kotlin

```kotlin
public external abstract class SVGAnimatedLengthList {
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthList
}
```

Exposes the JavaScript [SVGUnitTypes](https://developer.mozilla.org/en/docs/Web/API/SVGUnitTypes) to Kotlin

```kotlin
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGUnitTypes
```
companion object {
    val SVG_UNIT_TYPE_UNKNOWN: Short
    val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
}

/**
 * Exposes the JavaScript
 * [SVGTests](https://developer.mozilla.org/en/docs/Web/API/SVGTests) to Kotlin
 */
public external interface SVGTests {
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
}

public external interface SVGFitToViewBox {
    val viewBox: SVGAnimatedRect
    val preserveAspectRatio: SVGAnimatedPreserveAspectRatio
}

/**
 * Exposes the JavaScript
 * [SVGZoomAndPan](https://developer.mozilla.org/en/docs/Web/API/SVGZoomAndPan) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGZoomAndPan {
    var zoomAndPan: Short

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
    }
}

/**
 * Exposes the JavaScript
 * [SVGGElement](https://developer.mozilla.org/en/docs/Web/API/SVGGElement) to Kotlin
 */
public abstract class SVGGElement : SVGGraphicsElement, SVGFitToViewBox, SVGZoomAndPan, WindowEventHandlers {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open var currentScale: Float
    open var currentTranslate: DOMPointReadOnly
    fun getIntersectionList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun getEnclosureList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun deselectAll()
    fun createSVGNumber(): SVGNumber
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGMatrix(): DOMMatrix
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
    fun getElementById(elementId: String): Element
    fun suspendRedraw(maxWaitMilliseconds: Int): Int
    fun unsuspendRedraw(suspendHandleId: Int): Int
    fun unsuspendRedrawAll(): Unit
}

/**
 * Exposes the JavaScript
 * [SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin
 */
public abstract class SVGSVGElement : SVGGraphicsElement, SVGFitToViewBox, SVGZoomAndPan, WindowEventHandlers {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open var currentScale: Float
    open var currentTranslate: DOMPointReadOnly
    fun getIntersectionList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun getEnclosureList(rect: DOMRectReadOnly, referenceElement: SVGElement?): NodeList
    fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun deselectAll()
    fun createSVGNumber(): SVGNumber
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGMatrix(): DOMMatrix
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
    fun getElementById(elementId: String): Element
    fun suspendRedraw(maxWaitMilliseconds: Int): Int
    fun unsuspendRedraw(suspendHandleId: Int): Int
    fun unsuspendRedrawAll(): Unit
}

Open Source Used In DNAC Platform Arica 9939
Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin *
 */
public external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin *
 */
public external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val instanceRoot: SVGElement?
    open val animatedInstanceRoot: SVGElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external open class SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external interface SVGElementInstance {
    val correspondingElement: SVGElement?
        get() = definedExternally
    val correspondingUseElement: SVGUseElement?
        get() = definedExternally
}

public external open class ShadowAnimation(source: dynamic, newTarget: dynamic) {
    open val sourceAnimation: dynamic
}

/**
 * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin *
 */
public external abstract class SVGSwitchElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
Exposes the JavaScript [SVGAnimatedPreserveAspectRatio](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPreserveAspectRatio) to Kotlin */public abstract class SVGAnimatedPreserveAspectRatio {
    open val baseVal: SVGPreserveAspectRatio
    open val animVal: SVGPreserveAspectRatio
}

Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin */public abstract class SVGPathElement : SVGGeometryElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

Exposes the JavaScript [SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin */public abstract class SVGCircleElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin
 */
public external abstract class SVGLineElement : SVGGeometryElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
 */
public external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin
 */
public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
    fun removeItem(index: Int): DOMPoint
    fun appendItem(newItem: DOMPoint): DOMPoint
    fun getItem(index: Int): DOMPoint
}

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin
 */
public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {

Open Source Used In DNAC Platform Arica 9944
companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript
 [SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin
 */
public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript
 [SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextContentElement) to Kotlin
 */
public external abstract class SVGTextContentElement : SVGGraphicsElement {
    open val textLength: SVGAnimatedLength
    open val lengthAdjust: SVGAnimatedEnumeration
    fun getNumberOfChars(): Int
    fun getComputedTextLength(): Float
    fun getSubStringLength(charnum: Int, nchars: Int): Float
    fun getStartPositionOfChar(charnum: Int): DOMPoint
    fun getEndPositionOfChar(charnum: Int): DOMPoint
    fun getExtentOfChar(charnum: Int): DOMRect
    fun getRotationOfChar(charnum: Int): Float
    fun getCharNumAtPosition(point: DOMPoint): Int
    fun selectSubString(charnum: Int, nchars: Int)

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript
 [SVGTextPositioningElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPositioningElement) to
 Kotlin
 */
public external abstract class SVGTextPositioningElement : SVGTextContentElement {
    open val x: SVGAnimatedLengthList
    open val y: SVGAnimatedLengthList
    open val dx: SVGAnimatedLengthList
    open val dy: SVGAnimatedLengthList
    open val rotate: SVGAnimatedNumberList

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
/**
 * Exposes the JavaScript [SVGImageElement](https://developer.mozilla.org/en/docs/Web/API/SVGImageElement) to Kotlin
 */
public external abstract class SVGImageElement : SVGGraphicsElement, SVGURIReference,
HTMLOrSVGImageElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val preserveAspectRatio:
    SVGAnimatedPreserveAspectRatio
    open var crossOrigin: String?

    companion object {
        val
        ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE:
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGForeignObjectElement](https://developer.mozilla.org/en/docs/Web/API/SVGForeignObjectElement) to Kotlin
 */
public external abstract class SVGForeignObjectElement : SVGGraphicsElement {
    open val x:
    SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val
    height: SVGAnimatedLength

    companion object {
        val
        ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE:
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSolidcolorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement) to Kotlin
 */
public external abstract class SVGSolidcolorElement : SVGElement {
    companion object {
        val
        ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE:
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
**Exposes the JavaScript [SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin**

```kotlin
open val gradientUnits: SVGAnimatedEnumeration
open val gradientTransform: SVGAnimatedTransformList

companion object {
    val SVG_SPREADMETHOD_UNKNOWN: Short
    val SVG_SPREADMETHOD_PAD: Short
    val SVG_SPREADMETHOD_REFLECT: Short
    val SVG_SPREADMETHOD_REPEAT: Short
    val SVG_UNIT_TYPE_UNKNOWN: Short
    val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to Kotlin
 */
public external abstract class SVGLinearGradientElement : SVGGradientElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [SVGRadialGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGRadialGradientElement) to Kotlin
 */
public external abstract class SVGRadialGradientElement : SVGGradientElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength
    open val fx: SVGAnimatedLength
    open val fy: SVGAnimatedLength
    open val fr: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```
/**
 * Exposes the JavaScript [SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin *
 */
public
    abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLOrSVGScriptElement {
        open var type: String
        open var crossOrigin: String?

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }

/**
 * Exposes the JavaScript [SVGAElement](https://developer.mozilla.org/en/docs/Web/API/SVGAElement) to Kotlin *
 */
public
    abstract class SVGAElement : SVGGraphicsElement, SVGURIReference {
        open val target: SVGAAnimatedString
        open val download: SVGAAnimatedString
        open val rel: SVGAAnimatedString
        open val relList: SVGAAnimatedString
        open val hreflang: SVGAAnimatedString
        open val type: SVGAAnimatedString

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }

/**
 * Exposes the JavaScript [SVGViewElement](https://developer.mozilla.org/en/docs/Web/API/SVGViewElement) to Kotlin *
 */
public
    abstract class SVGViewElement : SVGElement, SVGFitToViewBox, SVGZoomAndPan {
        companion object {
            val SVG_ZOOMANDPAN_UNKNOWN: Short
            val SVG_ZOOMANDPAN_DISABLE: Short
            val SVG_ZOOMANDPAN_MAGNIFY: Short
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val SVG_ZOOMANDPAN_UNKNOWN: Short
            val SVG_ZOOMANDPAN_DISABLE: Short
            val SVG_ZOOMANDPAN_MAGNIFY: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }

/*
 * Copyright 2010-2020
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * NOTE: THIS FILE IS AUTO-
**Open Source Used In DNAC Platform Arica**

GENERATED, DO NOT EDIT!\n// See github.com/kotlin/dukat for details\n
```kotlin
package org.w3c.files

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.encryptedmedia.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.mediasource.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript
 [Blob](https://developer.mozilla.org/en/docs/Web/API/Blob) to Kotlin
 */
public external open class Blob(blobParts: Array<dynamic> = definedExternally, options: BlobPropertyBag = definedExternally) :
 MediaProvider, ImageBitmapSource {
    open val size: Number
    open val type: String
    open val isClosed: Boolean
    fun slice(start: Int = definedExternally, end: Int = definedExternally, contentType: String = definedExternally): Blob
    fun close()
}

public external interface BlobPropertyBag {
    var type: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun BlobPropertyBag(type: String? = ": BlobPropertyBag {\n    val o = js("({})")
    o["type"] = type
    return o
}**

/**
 * Exposes the JavaScript
 [File](https://developer.mozilla.org/en/docs/Web/API/File) to Kotlin
 */
public external open class File(fileBits: Array<dynamic>, fileName: String, options: FilePropertyBag = definedExternally) : Blob {
    open val name: String
    open val lastModified: Int
}

public external interface FilePropertyBag : BlobPropertyBag {
    var lastModified: Int?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun FilePropertyBag(lastModified: Int? = undefined, type: String? = ": FilePropertyBag {\n    val o = js("({})")
    o["lastModified"] = lastModified
    o["type"] = type
    return o
}**

/**
 * Exposes the JavaScript
 [FileList](https://developer.mozilla.org/en/docs/Web/API/FileList) to Kotlin
 */
public external abstract class FileList : ItemArrayLike<File> {
    override fun item(index: Int): File?
}

@kotlin.internal.InlineOnly
public operator fun FileList.get(index: Int): File? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [FileReader](https://developer.mozilla.org/en/docs/Web/API/FileReader) to Kotlin
 */
public external open class FileReader : EventTarget {
    open val readyState: Short
    open val result: dynamic
    open val error: dynamic
    var onloadstart: ((ProgressEvent) -> dynamic)?
    var onprogress: ((ProgressEvent) -> dynamic)?
    var onload: ((Event) -> dynamic)?
    var onabort: ((Event) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    fun readAsArrayBuffer(blob: Blob)
    fun readAsBinaryString(blob: Blob)
    fun readAsText(blob: Blob, label: String = definedExternally)
    fun readAsDataURL(blob: Blob)
    fun abort()\n
    companion object {
        val EMPTY: Short
        val LOADING: Short
        val DONE: Short
    }
}

/**
 * Exposes the JavaScript
 [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}
```

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

var notification: Notification?
notification.get() = definedExternally
set(value) = definedExternally

var action: String?
action.get() = definedExternally
set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["notification"] = notification
    o["action"] = action
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationPermission {
    companion object
}

public inline val NotificationPermission.Companion.DEFAULT: NotificationPermission get() = "default".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.DENIED: NotificationPermission get() = "denied".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.GRANTED: NotificationPermission get() = "granted".asDynamic().unsafeCast<NotificationPermission>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationDirection {
    companion object
}

public inline val NotificationDirection.Companion.AUTO: NotificationDirection get() = "auto".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.LTR: NotificationDirection get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.RTL: NotificationDirection get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details
//
Boolean? = false, type: ClientType? = ClientType.WINDOW): ClientQueryOptions {
    val o = js("({})")
    o["includeUncontrolled"] = includeUncontrolled
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript
 */
public external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {
    fun waitUntil(f: Promise<Any?>)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ExtendableEventInit : EventInit

@kotlin.internal.InlineOnly
public inline fun ExtendableEventInit(bubbles: Boolean? = false, cancelable: Boolean? = false): ExtendableEventInit {
    val o = js("({})")
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 */
public external open class InstallEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : ExtendableEvent {
    fun registerForeignFetch(options: ForeignFetchOptions)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ForeignFetchOptions {
    var scopes: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var origins: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ForeignFetchOptions(scopes: Array<String>?, origins: Array<String>?): ForeignFetchOptions {
    val o = js("({})")
    o["scopes"] = scopes
    o["origins"] = origins
    return o
}

/**
 * Exposes the JavaScript
 */
public external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {
    open val request: Request
    open val clientId: String?
    open val isReload: Boolean
    fun respondWith(r: Promise<Response>)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface FetchEventInit : ExtendableEventInit {
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var clientId: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var isReload: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["request"] = request
    o["clientId"] = clientId
    o["isReload"] = isReload
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external interface ForeignFetchResponse {
    var response: Response?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
    var headers: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["response"] = response
    o["origin"] = origin
    o["headers"] = headers
    return o
}

/**
 * Exposes the JavaScript
 */

Open Source Used In DNAC Platform Arica 9956
Open Source Used In DNAC Platform Arica 9957

[ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin

```kotlin
public external open class ExtendableMessageEvent(type: String, eventInitDict: ExtendableMessageEventInit = definedExternally) : ExtendableEvent {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val open value: UnionClientOrMessagePortOrServiceWorker?
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ExtendableMessageEventInit : ExtendableEventInit {
    var data: Any?
        set(value) = definedExternally
    var origin: String?
        set(value) = definedExternally
    var lastEventId: String?
        set(value) = definedExternally
    var source: UnionClientOrMessagePortOrServiceWorker?
        set(value) = definedExternally
    var ports: Array<MessagePort>?
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ExtendableMessageEventInit(data: Any? = undefined, origin: String? = undefined, lastEventId: String? = undefined, 
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
    @n@kotlin.internal.InlineOnly
}
```

[Cache](https://developer.mozilla.org/en/docs/Web/API/Cache) to Kotlin

```kotlin
public external abstract class Cache {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally): Promise<Array<Response>>
    fun add(request: dynamic): Promise<Unit>
    fun addAll(requests: Array<dynamic>): Promise<Unit>
    fun put(request: dynamic, response: Response): Promise<Unit>
    fun delete(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>
    fun keys(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally): Promise<Array<Request>>
}

public external interface CacheQueryOptions {
    var ignoreSearch: Boolean? /* = false */
    var ignoreMethod: Boolean? /* = false */
    var ignoreVary: Boolean? /* = false */
    var cacheName: String?
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ignoreSearch"] = ignoreSearch
    o["ignoreMethod"] = ignoreMethod
    o["ignoreVary"] = ignoreVary
    o["cacheName"] = cacheName
    return o
}
```

[CacheBatchOperation](https://developer.mozilla.org/en/docs/Web/API/CacheBatchOperation) to Kotlin

```kotlin
public external interface CacheBatchOperation {
    var type: String?
    var request: Request?
    var response: Response?
    var options: CacheQueryOptions?
}

@kotlin.internal.InlineOnly
options: CacheQueryOptions? = undefined): CacheBatchOperation {
    val o = js("({})")
    o["type"] = type
    o["request"] = request
    o["response"] = response
    o["options"] = options
    return o
}
```

[CacheStorage](https://developer.mozilla.org/en/docs/Web/API/CacheStorage) to Kotlin

```kotlin
public external abstract class CacheStorage {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun has(cacheName: String): Promise<Boolean>
    fun open(cacheName: String): Promise<Cache>
    fun delete(cacheName: String): Promise<Boolean>
    fun keys(): Promise<Array<String>>
}
```

[FunctionalEvent](https://developer.mozilla.org/en/docs/Web/API/FunctionalEvent) to Kotlin

```kotlin
public external open class FunctionalEvent : ExtendableEvent {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}
```
```kotlin
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ServiceWorkerState {
    companion object
}

public inline val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() = "installing".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() = "installed".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() = "activating".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() = "activated".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.REDUNDANT: ServiceWorkerState get() = "redundant".asDynamic().unsafeCast<ServiceWorkerState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface FrameType {
    companion object
}

public inline val FrameType.Companion.AUXILIARY: FrameType get() = "auxiliary".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.TOP_LEVEL: FrameType get() = "top-level".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NESTED: FrameType get() = "nested".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NONE: FrameType get() = "none".asDynamic().unsafeCast<FrameType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ClientType {
    companion object
}

public inline val ClientType.Companion.WINDOW: ClientType get() = "window".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.WORKER: ClientType get() = "worker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.SHAREDWORKER: ClientType get() = "sharedworker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.ALL: ClientType get() = "all".asDynamic().unsafeCast<ClientType>()

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. 

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details

// import statements
```

Open Source Used In DNAC Platform Arica 9958
fun setRequestHeader(name: String, value: String)
fun send(body: dynamic = definedExternally)
fun abort()
fun getResponseHeader(name: String): String?
fun getAllResponseHeaders(): String
fun overrideMimeType(mime: String)

companion object {
    val UNSENT: Short
    val OPENED: Short
    val HEADERS_RECEIVED: Short
    val LOADING: Short
    val DONE: Short
}

/**
 * Exposes the JavaScript
 * [FormData](https://developer.mozilla.org/en/docs/Web/API/FormData) to Kotlin
 */
public external open class FormData(form: HTMLFormElement = definedExternally) {
    fun append(name: String, value: String)
    fun append(name: String, value: Blob, filename: String = definedExternally)
    fun delete(name: String)
    fun get(name: String): dynamic
    fun getAll(name: String): Array<dynamic>
    fun has(name: String): Boolean
    fun set(name: String, value: String)
    fun set(name: String, value: Blob, filename: String = definedExternally)
}

/**
 * Exposes the JavaScript
 * [ProgressEvent](https://developer.mozilla.org/en/docs/Web/API/ProgressEvent) to Kotlin
 */
public external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) : Event {
    open val lengthComputable: Boolean
    open val loaded: Number
    open val total: Number

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ProgressEventInit : EventInit {
    var lengthComputable: Boolean? /* = false */
    var loaded: Number? /* = 0 */
    var total: Number? /* = 0 */
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["lengthComputable"] = lengthComputable
    o["loaded"] = loaded
    o["total"] = total
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface XMLHttpRequestResponseType {
    companion object
}

public inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() = "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() = "arraybuffer".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = "blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = "document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = "json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = "text".asDynamic().unsafeCast<XMLHttpRequestResponseType>()
/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationTarget.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Allows to infer generic type arguments of a function from the calls in the annotated function parameter of that function.
 *
 * When this annotation is placed on a generic function parameter of a function, it enables to infer the type arguments of that generic function from the lambda body passed to that parameter. The calls that affect inference are either members of the receiver type of an annotated function parameter or extensions for that type. The extensions must be themselves annotated with `@BuilderInference`. Example: we declare:

```kotlin
fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T>
```

and use it like:

```kotlin
val result = sequence {
    yield("result")
}
```

Here the type argument of the resulting sequence is inferred to `String` from the argument of the `SequenceScope.yield` function, that is called inside the lambda passed to the `Sequence`. Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.

*/
@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3.3")
@ExperimentalTypeInference
public annotation class BuilderInference

/**
 * Enables overload selection based on the type of the value
returned from lambda argument.

* When two or more function overloads have otherwise the same parameter lists that differ only in the return type of a functional parameter, this annotation enables overload selection by the type of the value returned from the lambda function passed to this functional parameter.

* Example:

```kotlin
@OverloadResolutionByLambdaReturnType
fun create(intProducer: () -> Int): Int

fun create(doubleProducer: () -> Double): Double

val newValue = create { 3.14 }
```

This annotation is also used to discriminate the annotated overloads in case if overload selection still cannot choose one of them even taking in account the result of lambda parameter analysis. In that case a warning is reported.

* Note: this annotation is experimental, see [OverloadResolutionByLambdaReturnType] on how to opt-in for it.

```kotlin
@Target(FUNCTION)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.4")
@Experimental
@Public
public annotation class OverloadResolutionByLambdaReturnType
```

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

---

**Open Source Used In DNAC Platform Arica**

9961

---

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
Open Source Used In DNAC Platform Arica 9962

@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70\n", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class RequiresOptIn(\n    val message: String = "",
    val level: Level = Level.ERROR
) {
    /**
     * Severity of the diagnostic that should be reported on usages which did not explicitly opted into
     * the API either by using [OptIn] or by being annotated with the corresponding marker annotation.
     * @ \n    * \n     * \n    */
    public enum class Level {
        /** Specifies that an error should be reported on incorrect usages of this API. */
        ERROR,

        /**
         * WARNING,\n         */
        WARNING,
    }
}

/**
 * Allows to use the API denoted by the given markers in the annotated file, declaration, or expression.
 * @ \n    * \n     * \n    */
@Target(
    CLASS, PROPERTY,
    LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER,
    PROPERTY_SETTER, EXPRESSION, FILE,
)
@Retention(SOURCE)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class OptIn(\n    vararg val markerClass: KClass<out Annotation>
)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.collections

import kotlin.js.JsName

/**
 * Provides a skeletal implementation of the read-only [Collection] interface.
 *
 * @param E the type of elements contained in the collection. The collection is covariant in its element type.
 */
@SinceKotlin("1.1")
public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }
    override fun override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean = this.elements.all { contains(it) }

    override fun isEmpty(): Boolean = size == 0

    /**
     * Returns new array of type `Array<Any?>` with the elements of this collection.
     */
    @JsName("toArray")
    protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

    /**
     * Fills the provided `array` or creates new array of the same type and fills it with the elements of this collection.
     */
    protected open fun <T> toArray(array: Array<T>): Array<T> = copyToArrayImpl(this, array)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

private enum class State {
    Ready,
    NotReady,
    Done,
    Failed
}

/**
 * A base class to simplify implementing iterators so that implementations only have to implement [computeNext] to implement the iterator, calling [done] when the iteration is complete.
 * @param E the type of elements contained in the collection.
 */
public abstract class AbstractIterator<T> : Iterator<T> {
    private var state = State.NotReady
    private var nextValue: T? = null

    override fun hasNext(): Boolean = when (state) {
        State.Done -> false
        State.Failed -> true
        else -> tryToComputeNext() != null
    }

    override fun next(): T = if (hasNext()) nextValue else throw NoSuchElementException()

    /**
     * Computes the next value in the iteration, called from the [computeNext] function
     * @param E the type of elements contained in the collection.
     * @return the next value or null if the iteration terminates with a failed state
     */
    protected fun computeNext(): T = if (hasNext()) state

    /**
     * This callback method should call one of these two methods:
     * @param E the type of elements contained in the collection.
     * @param done if the iteration terminates with a failed state
     * @param E the type of elements contained in the collection.
     * @param done if the iteration terminates with a failed state
     */
    protected fun setNext(value: T): Unit = Unit

    /**
     * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
     * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
     */
}
public fun lastOrNull(): E? = if (isEmpty()) null else internalGet(internalIndex(lastIndex))

private fun copyCollectionElements(internalIndex: Int, elements: Collection<E>) {
    val iterator = elements.iterator()
    if (iterator.hasNext()) {
        val element = iterator.next()
        if (elementData.size == 1) {
            elementData[0] = element
        } else {
            val tail = internalIndex(size)
            if (decrementedInternalIndex >= head) {
                elementData[decrementedHead] = element
                head = incremented(head)
                size -= 1
                return element
            } else {
                elementData.copyInto(elementData, head + 1)
                if (index == 0) {
                    addFirst(element)
                    return true
                }

                ensureCapacity(size + 1)
                val element = internalGet(internalLastIndex)
                val elementData = internalData.copyInto(elementData)
                if (index == size) {
                    addLast(element)
                    return true
                }

                ensureCapacity(size + 1)
                val head = incremented(head)
                elementData[head] = element
                size += 1
                return element
            }
        }
    }

    public override fun add(index: Int, element: E) {
        AbstractList.checkPositionIndex(index, size)
        if (index == size) {
            addLast(element)
            return true
        }

        ensureCapacity(size + 1)
        val element = internalGet(internalIndex(lastIndex))
        val elementData = internalData.copyInto(elementData)
        if (index == 0) {
            addFirst(element)
            return true
        }

        ensureCapacity(size + 1)
        val head = incremented(head)
        elementData[head] = element
        size += 1
        return element
    }

    /**
     * Adds the specified [element] to the end of this deque.
     */
    public fun addLast(element: E) {
        ensureCapacity(size + 1)
        val element = internalGet(internalLastIndex)
        val elementData = internalData.copyInto(elementData)
        if (index == size) {
            addLast(element)
            return true
        }

        ensureCapacity(size + 1)
        val head = incremented(head)
        elementData[head] = element
        size += 1
        return element
    }

    /**
     * Adds the specified [element] to the beginning of this deque.
     */
    public fun addFirst(element: E) {
        ensureCapacity(size + 1)
        val element = internalGet(head)
        val elementData = internalData.copyInto(elementData, head + 1)
        if (index == 0) {
            addFirst(element)
            return true
        }

        ensureCapacity(size + 1)
        val head = incremented(head)
        elementData[head] = element
        size += 1
        return element
    }

    /**
     * Removes the last element from this deque and returns that removed element, or throws [NoSuchElementException] if this deque is empty.
     */
    public fun removeLast(): E {
        if (isEmpty()) throw NoSuchElementException("ArrayDeque is empty.")
        val internalLastIndex = internalIndex(lastIndex)
        val element = internalGet(internalLastIndex)
        val elementData = internalData.copyInto(elementData, internalLastIndex)
        if (index == size) {
            addLast(element)
            return element
        }

        ensureCapacity(size + 1)
        val head = incremented(head)
        elementData[head] = null
        head = incremented(head)
        size -= 1
        return element
    }

    /**
     * Removes the first element from this deque and returns that removed element, or throws `null` if this deque is empty.
     */
    public fun removeFirst(): E? = if (isEmpty()) null else removeFirst()

    /**
     * Removes the last element from this deque and returns that removed element, or throws [NoSuchElementException] if this deque is empty.
     */
    public fun removeLastOrNull(): E? = if (isEmpty()) null else removeLast()

    /**
     * Removes the first element from this deque and returns that removed element, or throws `null` if this deque is empty.
     */
    public fun removeFirstOrNull(): E? = if (isEmpty()) null else removeFirst()
elements.iterator() for (index in internalIndex until elementData.size) {
    if (!iterator.hasNext())
        break
    elementData[index] = iterator.next()
} for (index in 0 until head) {
    if (!iterator.hasNext())
        break
    elementData[index] = iterator.next()
} size += elements.size
}

public override fun addAll(elements: Collection<E>): Boolean {
    if (elements.isEmpty()) return false
    ensureCapacity(this.size + elements.size)
    copyCollectionElements(internalIndex(size), elements)
    return true
}

public override fun addAll(index: Int, elements: Collection<E>): Boolean {
    if (index == size) {
        return addAll(elements)
    } else if (index >= size) {
        return addAll(elements)
    } else
        return false
}

ensureCapacity(this.size + elements.size)

val tail = internalIndex(size)
val internalIndex = internalIndex(index)
val elementsSize = elements.size
val shiftedHead = head - elementsSize
val internalIndex = internalIndex(index)
val tail = internalIndex(size)
val internalIndex = internalIndex(index)
val elementsSize = elements.size

if (index == size) {
    return addAll(elements)
} else
    return false

var shiftedHead = head - elementsSize
val internalIndex = internalIndex(index)
val elementsSize = elements.size

if (index >= size) {
    return addAll(elements)
} else
    return false

ensureCapacity(this.size + elements.size)

copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)
}

if (elements.isEmpty()) {
    return false
} else if (index == size) {
    return addAll(elements)
} else {
    ensureCapacity(this.size + elements.size)
    copyCollectionElements(internalIndex(size), elements)
}

if (index == size) {
    return addAll(elements)
} else if (index >= size) {
    return addAll(elements)
} else
    return false

ensureCapacity(this.size + elements.size)

copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)
}

if (elements.isEmpty()) {
    return false
} else if (index == size) {
    return addAll(elements)
} else {
    ensureCapacity(this.size + elements.size)
    copyCollectionElements(internalIndex(size), elements)
}

if (index == size) {
    return addAll(elements)
} else if (index >= size) {
    return addAll(elements)
} else
    return false

ensureCapacity(this.size + elements.size)

copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)
}

if (elements.isEmpty()) {
    return false
} else if (index == size) {
    return addAll(elements)
} else {
    ensureCapacity(this.size + elements.size)
    copyCollectionElements(internalIndex(size), elements)
}

if (index == size) {
    return addAll(elements)
} else if (index >= size) {
    return addAll(elements)
} else
    return false

ensureCapacity(this.size + elements.size)

copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)
}

return false

if (index == size) {
    return addAll(elements)
} else if (index >= size) {
    return addAll(elements)
} else
    return false

ensureCapacity(this.size + elements.size)

copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)
}
else {
    elementData.copyInto(elements, startVertex = head)
    elementData.copyInto(elements, elementData.size - head, endVertex = tail)
}
}
}

Array<out T>.flattenToArray(): Array<out T> *

public fun <T> Array<out T>.flatten(): Array<out T> {
    val result = ArrayList<out T>(size)
    for (i in indices) {
        result.add(All<out T>(this)
                .sumBy { it.size })
    }
    return result
}

Array<out T>.zip(): Array<out Pair<T, R>> *

public fun <T, R> Array<out Pair<T, R>>.zip(): Array<out Pair<T, R>> {
    val listT = ArrayList<T>(size)
    val listR = ArrayList<R>(size)
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

Array<*>?.isNullOrEmpty(): Boolean *

public inline fun Array<*>?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || empty()
}

Array<*>?.ifEmpty(defaultValue: () -> R): R *

public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Array<*>, C : R =
    if (isEmpty())
        defaultValue() else
    this
}

fun <T> Array<out T>?.contentDeepEqualsImpl(other: Array<out T>?): Boolean {
    if (this === other) return true
    if (this == null || other == null || this.size != other.size) return false

    for (i in indices) {
        val v1 = this[i]
        val v2 = other[i]

        if (v1 === v2) {
            continue
        } else if (v1 == null || v2 == null) {
            return false
        }

        when {
            v1 is Array<*> && v2 is Array<*> -> if (!v1.contentDeepEquals(v2)) return false
            v1 is ByteArray && v2 is ByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is ShortArray && v2 is ShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is IntArray && v2 is IntArray -> if (!v1.contentEquals(v2)) return false
            v1 is LongArray && v2 is LongArray -> if (!v1.contentEquals(v2)) return false
            v1 is FloatArray && v2 is FloatArray -> if (!v1.contentEquals(v2)) return false
            v1 is DoubleArray && v2 is DoubleArray -> if (!v1.contentEquals(v2)) return false
            v1 is CharArray && v2 is CharArray -> if (!v1.contentEquals(v2)) return false
            v1 is BooleanArray && v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false
            v1 is UByteArray && v2 is UByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is UShortArray && v2 is UShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is UIntArray && v2 is UIntArray -> if (!v1.contentEquals(v2)) return false
            v1 is ULongArray && v2 is ULongArray -> if (!v1.contentEquals(v2)) return false
            else -> if (v1 != v2) return false
        }
    }
    return true
}
is missing in the map.\n
```
mapWithDefault(defaultValue: (key: K) -> V): MapWithDefaultImpl<K, V> =
    when (this) {
        is MapWithDefault -> this.mapWithDefault(defaultValue) else ->
            MapWithDefaultImpl(this, defaultValue)
    }

MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =
    when (this) {
        is MutableMapWithDefault -> this.mapWithDefault(defaultValue) else ->
            MutableMapWithDefaultImpl(this, defaultValue)
    }
```

```
private interface MapWithDefault<K, out V> : Map<K, V> {
    public val map: Map<K, V>
    public fun getOrImplicitDefault(key: K): V
}

private interface MutableMapWithDefault<K, V> :
    MutableMap<K, V>, MapWithDefault<K, V> {
    public override val map: MutableMap<K, V>
}

private class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>,
    private val default: (key: K) -> V) : MapWithDefault<K, V> {
    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean =
        map.containsValue(value)
    override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, {
        default(key) })
}
```

```
private class MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K,
    V>, private val default: (key: K) -> V) :
    MutableMapWithDefaultImpl<K, V>, private val default: (key: K) -> V) :
    MutableMapWithDefaultImpl<K, V> {
    override fun put(key: K, value: V): V? =
        map.put(key, value)
    override fun putAll(from: Map<out K, V>) =
        map.putAll(from)
    override fun forEach(block: (key: K, value: V) -> Unit) =
        map.forEach(block)
    override fun remove(key: K): V? = map.remove(key)
    override fun putIfAbsent(key: K, value: V): Boolean =
        map.putIfAbsent(key, value)
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean =
        map.containsValue(value)
    override fun getOrImplicitDefault(key: K): V =
        map.getOrElseNullable(key, {
            default(key) })
    override fun equals(other: Any?): Boolean =
        map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean =
        map.containsValue(value)
    override fun getOrImplicitDefault(key: K): V =
        map.getOrElseNullable(key, {
            default(key) })
}
```

```
public inline fun <T> MutableCollection<T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CASTV") (this as Collection<T>).remove(element)
```

```
public inline fun MutableList<T>.removeAll(from: Collection<T>): Boolean =
    this.removeAll(from)
```
MutableCollection\<out T\>.removeAll(elements: Collection\<T\>): Boolean

`@Suppress("UNCHECKED_CAST")` (this as MutableCollection\<T\>).removeAll(elements)\n\nRetains only the elements in this collection that are contained in the specified collection.\n\nAllows to overcome type-safety restriction of `retainAll` that requires to pass a collection of type `Collection`.

When `true` if any element was removed from the collection, `false` if the collection was not modified.

`@kotlin.internal.InlineOnly\!/public inline fun <T> MutableCollection\<out T\>.retainAll(elements: Collection\<T\>): Boolean`
MutableCollection<in T>.retainAll(elements: Sequence<T>): Boolean {
    val set = elements.toHashSet()
    if (set.isNotEmpty())
        return retainAll(set)
    else
        return retainNothing()
}

private fun <T> MutableCollection<in T>.retainNothing(): Boolean {
    val result = isNotEmpty()
    clear()
    return result
}

/**
 * Removes all elements from this [MutableIterable] that match the given [predicate].
 * @return `true` if any element was removed from this collection, or `false` when no elements were removed and collection
 * was not modified.
 */
public fun <T> MutableIterable<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)

/**
 * Retains only elements of this [MutableIterable] that match the given [predicate].
 * @return `true` if any element was removed from this collection, or `false` when all elements were retained and
collection was not modified.
 */
public fun <T> MutableIterable<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)

private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    if (this !is RandomAccess)
        return (this as MutableIterable<T>).filterInPlace(predicate, predicateResultToRemove)

    var writeIndex: Int = 0
    for (readIndex in 0..lastIndex) {
        val element = this[readIndex]
        if (predicate(element) == predicateResultToRemove)
            continue

        if (writeIndex != readIndex)
            this[writeIndex] = element

        writeIndex++
    }
    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo writeIndex)
            removeAt(removeIndex)

        return true
    } else {
        return false
    }
}

package
private open class ReversedListReadOnly<out T>(private val delegate: List<T>) : AbstractList<T>() {
    override val size: Int get() = delegate.size
    override fun get(index: Int): T = delegate[reverseElementIndex(index)]
}

private class ReversedList<T>(private val delegate: MutableList<T>) : AbstractMutableList<T>() {
    override val size: Int get() = delegate.size
    override fun get(index: Int): T = delegate[reverseElementIndex(index)]
    override fun clear() = delegate.clear()
    override fun removeAt(index: Int): T = delegate.removeAt(reverseElementIndex(index))

    override fun set(index: Int, element: T): T = delegate.set(reverseElementIndex(index), element)
    override fun add(index: Int, element: T) {
        delegate.add(reversePositionIndex(index), element)
    }
}

private fun List<*>.reverseElementIndex(index: Int) =
    if (index in 0..lastIndex) lastIndex - index else throw IndexOutOfBoundsException("Element index $index must be in range [0..lastIndex].")

private fun List<*>.reversePositionIndex(index: Int) =
    if (index in 0..size) size - index else throw IndexOutOfBoundsException("Position index $index must be in range [0..size].")

/**
 * Returns a reversed read-only view of the original List.
 * All changes made in the original list will be reflected in the reversed one.
 * @sample samples.collections.ReversedViews.asReversedList
 */
public fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this)

/**
 * Returns a reversed mutable view of the original mutable List.
 * All changes made in the original list will be reflected in the reversed one and vice versa.
 * @sample samples.collections.ReversedViews.asReversedMutableList
 */
@kotlin.jvm.JvmName("asReversedMutable")
public fun <T> MutableList<T>.asReversed(): MutableList<T> = ReversedList(this)

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")
@file:OptIn(ExperimentalTypeInference::class)

package kotlin.sequences

import kotlin.coroutines.*
import kotlin.coroutines.intrinsics.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Builds a lazyly yielding values one by one.
 * @see kotlin.sequences.generateSequence
 * @sample samples.collections.Sequences.buildSequenceYieldAll
 * @sample samples.collections.Sequences.buildFibonacciSequence
 */
@SinceKotlin("1.3")
public fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(block)
}

@SinceKotlin("1.3")
@Deprecated("Use 'sequence { }' function instead.",
    ReplaceWith("sequence(builderAction)"), level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> =
    Sequence {
        iterator(builderAction)
    }

/**
 * Builds an [Iterator] lazily yielding values one by one.
 * @sample samples.collections.Sequences.buildFibonacciSequence
 **/
@SinceKotlin("1.3")
public fun <T> iterator(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Iterator<T> {
    val iterator = SequenceBuilderIterator<T>()
    iterator.nextStep = block.createCoroutineUnintercepted(receiver = iterator, completion = iterator)
    return iterator
}

@SinceKotlin("1.3")
@Deprecated("Use 'iterator { }' function instead.",
    ReplaceWith("iterator(builderAction)"), level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> buildIterator(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Iterator<T> =
    iterator(builderAction)

/**
 * The scope for yielding values of a [Sequence] or an [Iterator],
 * provides [yield] and [yieldAll] suspension functions.
 * @see KotlinSequence
 * @sample samples.collections.Sequences.buildSequenceYieldAll
 **/
@SinceKotlin("1.3")
@RestrictsSuspension
public abstract class SequenceScope<in T> internal constructor() {
    /**
     * Yields a value to the [Iterator] being built and suspends
     * until the next value is requested.
     */
    override suspend fun yield(value: T): Unit =

    /**
     * Yields all values from the `iterator` to the [Iterator]
     * being built and suspends until all these
     */
}
values are iterated and the next one is requested.

* The sequence of values returned by the given iterator can be potentially infinite.
* The sequence of values returned by the given iterator can be potentially infinite.
* Yields a collection of values to the [Iterator] being built
* and suspends until all these values are iterated and the next one is requested.

* Yields potentially infinite sequence of values to the [Iterator] being built
* and suspends until all these values are iterated and the next one is requested.

* The sequence can be potentially infinite.

* @sample samples.collections.Sequences.Building.buildSequenceYieldAll

```kotlin
public abstract suspend fun yieldAll(iterator: Iterator<T>)

/**
 * Yields a collection of values to the [Iterator] being built
 * and suspends until all these values are iterated and the next one is requested.
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
*/
    public abstract suspend fun yieldAll(iterator: Iterator<T>)

    /**
     * Yields a collection of values to the [Iterator] being built
     * and suspends until all these values are iterated and the next one is requested.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public suspend fun yieldAll(elements: Iterable<T>) {
        if (elements is Collection && elements.isEmpty()) return
        return yieldAll(elements.iterator())
    }

    /**
     * Yields potentially infinite sequence of values to the [Iterator] being built
     * and suspends until all these values are iterated and the next one is requested.
     * The sequence can be potentially infinite.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public suspend fun yieldAll(sequence: Sequence<T>) = yieldAll(sequence.iterator())

    @Deprecated("Use SequenceScope class instead.", ReplaceWith("SequenceScope<T>"), level = DeprecationLevel.ERROR)
public typealias SequenceBuilder<T> = SequenceScope<T>

private typealias State = Int

private const val State_NotReady: State = 0
private const val State_ManyNotReady: State = 1
private const val State_ManyReady: State = 2
private const val State_Ready: State = 3
private const val State_Done: State = 4
private const val State_Failed: State = 5

private class SequenceBuilderIterator<T> : SequenceScope<T>(), Iterator<T>, Continuation<Unit> {
    private var state = State_NotReady
    private var nextValue: T? = null
    private var nextIterator: Iterator<T>? = null
    var nextStep: Continuation<Unit>? = null

    override fun hasNext(): Boolean {
        while (true) {
            when (state) {
                State_NotReady -> {}
                State_ManyNotReady ->
                    if (nextIterator!!.hasNext()) {
                        state = State_ManyReady
                        return true
                    } else {
                        nextIterator = null
                        state = State_Done
                    }
                State_Done -> return false
                State_Ready, State_ManyReady -> return true
                else -> throw exceptionalState()
            }

            state = State_Failed
            val step = nextStep!!
            nextStep = null
            step.resume(Unit)
        }
    }

    override fun next(): T {
        when (state) {
            State_NotReady, State_ManyNotReady -> return nextNotReady()
            State_ManyReady -> {
                state = State_ManyNotReady
                return nextIterator!!.next()
            }
            State_Ready -> {
                state = State_NotReady
                @Suppress("UNCHECKED_CAST")
                val result = nextValue as T
                nextValue = null
                return result
            }
            else -> throw exceptionalState()
        }
    }

    private fun nextNotReady(): T {
        if (!hasNext()) throw NoSuchElementException() else return next()
    }

    private fun exceptionalState(): Throwable = when (state) {
        State_Done -> NoSuchElementException()
        State_Failed
        -> IllegalStateException("Iterator has failed.")
        else -> IllegalStateException("Unexpected state of the iterator: $state")
    }

    override suspend fun yield(value: T) {
        nextValue = value
        state = State_Ready
        return suspendCoroutineUninterceptedOrReturn { c ->
            nextStep = c
            EXCEPTIONAL
        }
    }

    override suspend fun yieldAll(iterator: Iterator<T>) {
        if (!iterator.hasNext()) return
        nextIterator = iterator
        state = State_ManyReady
        return suspendCoroutineUninterceptedOrReturn { c ->
            nextStep = c
            EXCEPTIONAL
        }
    }

    // Completion continuation implementation
    override fun resumeWith(result: Result<Unit>) {
        result.getOrThrow() // just rethrow exception if it is there
        state = State_Done
        override val context: CoroutineContext
            get() = EmptyCoroutineContext
    }
    ```
inline fun Int.forward(n: Int): Int = (this + n) % capacity

fun UByteArray.

inline fun Int.forward(n: Int): Int = (this + n) % capacity

fun UByteArray.

private fun partition(array: UByteArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i < j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

private fun quickSort(array: UByteArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// Interfaces

/**
 * Sorts the given array using qsort algorithm.
 */
@ExperimentalUnsignedTypes
internal fun sortArray(array: UByteArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: UShortArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: UIntArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
@ExperimentalUnsignedTypes
internal fun sortArray(array: ULongArray, fromIndex: Int, toIndex: Int) = quickSort(array, fromIndex, toIndex - 1)
annotation, e.g. `@OptIn(ExperimentalContracts::class)` * or by using the compiler argument `-Xopt-in=kotlin.contracts.ExperimentalContracts` * Provides a scope, where the functions of the contract DSL, such as [returns], [callsInPlace], etc., can be used to describe the contract of a function. * This type is used as a receiver type of the lambda function passed to the [contract] function.

 */
@Suppress("DEPRECATION")
@Retention(AnnotationRetention.BINARY)
@Experimental
@RequiresOptIn
@MustBeDocumented
public annotation class ExperimentalContracts

/**
 * Provides a scope, where the functions of the contract DSL, such as [returns], [callsInPlace], etc., can be used to describe the contract of a function. * This type is used as a receiver type of the lambda function passed to the [contract] function.

 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public interface ContractBuilder {
    /**
     * Describes a situation when a function returns normally, without any exceptions thrown.
     */
    @ContractsDsl public fun returns(): Returns

    /**
     * Describes a situation when a function returns normally with the specified return value.
     */
    // @sample samples.contracts.returnsContract
    @ContractsDsl public fun returns(value: Any?): Returns

    /**
     * Describes a situation when a function returns normally with any value that is not `null`.
     */
    // @sample samples.contracts.returnsNotNullContract
    @ContractsDsl public fun returnsNotNull(): ReturnsNotNull

    /**
     * Specifies that the function parameter [lambda] is invoked in place.
     */
    // @sample samples.contracts.callsInPlaceAtMostOnceContract
    // @sample samples.contracts.callsInPlaceAtLeastOnceContract
    // @sample samples.contracts.callsInPlaceExactlyOnceContract
    // @sample samples.contracts.callsInPlaceUnknownContract
    @ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace
}

/**
 * Specifies how many times a function invokes its function parameter in place.
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public enum class InvocationKind {
    /**
     * A function parameter will be invoked one time or not invoked at all.
     */
    // @sample samples.contracts.callsInPlaceAtMostOnceContract
    @ContractsDsl AT_MOST_ONCE,

    /**
     * A function parameter will be invoked one or more times.
     */
    // @sample samples.contracts.callsInPlaceAtLeastOnceContract
    @ContractsDsl AT_LEAST_ONCE,

    /**
     * A function parameter will be invoked exactly one time.
     */
    // @sample samples.contracts.callsInPlaceExactlyOnceContract
    @ContractsDsl EXACTLY_ONCE,

    /**
     * A function parameter is called in place, but it’s unknown how many times it can be called.
     */
    // @sample samples.contracts.callsInPlaceUnknownContract
    @ContractsDsl UNKNOWN
}

/**
 * Specifies how many times a function invokes its function parameter in place.
 */
/* @sample samples.contracts.returnsContract
* @sample samples.contracts.returnsTrueContract
* @sample samples.contracts.returnsFalseContract
* @sample samples.contracts.returnsNullContract
* @sample samples.contracts.callsInPlaceAtMostOnceContract
* @sample samples.contracts.callsInPlaceAtLeastOnceContract
* @sample samples.contracts.callsInPlaceExactlyOnceContract
* @sample samples.contracts.callsInPlaceUnknownContract
*/

Open Source Used In DNAC Platform Arica 9978
Open Source Used In DNAC Platform Arica 9979
lambda creation
    context.fold(this) { acc, element ->
        val removed = acc.minusKey(element.key)
        if (removed == EmptyCoroutineContext) element else
        if (interceptor == null) CombinedContext(removed, element) else
            val left = removed.minusKey(ContinuationInterceptor)
        CombinedContext(CombinedContext(left), interceptor)
    } -

Returns a context containing elements from this context, but without an element with
* the specified [key].

public fun minusKey(key: Key<*>) =
    if (this.key == key) EmptyCoroutineContext else this

public interface Key<E : Element> -

An element of the [CoroutineContext]. An element of the coroutine context is a singleton context by itself.

public interface Element : CoroutineContext {
    public val key: Key<*>

    public override operator fun <E : Element> get(key: Key<E>): E? =
        if (this.key == key) this as E else
        public override fun <R> fold(initial: R, operation: (R, Element) -> R): R =
            operation(initial, this)

    public override
        fun minusKey(key: Key<*>): CoroutineContext =
            if (this.key == key) EmptyCoroutineContext else this

    // make sure interceptor is always last in the context (and thus is fast to get when present)
    val interceptor = removed[ContinuationInterceptor]
    if (interceptor == null) CombinedContext(removed, element) else
        val left = removed.minusKey(ContinuationInterceptor)
    CombinedContext(CombinedContext(left), interceptor)
}

package kotlin.coroutines

import kotlin.coroutines.CoroutineContext.Element
import kotlin.coroutines.CoroutineContext.Key

/**
 * Base class for [CoroutineContext.Element] implementations.
 */
@SinceKotlin("1.3")
public abstract class AbstractCoroutineContextElement(public override val key: Key<*>) : Element

/**
 * Base class for [CoroutineContext.Key] associated with polymorphic
 * [CoroutineContext.Element] implementation.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public abstract class AbstractCoroutineContextKey<B : Element, E : B>(
    baseKey: Key<B>,
    private val safeCast: (element: Element) -> E?
) : Key<E> {
    private val topmostKey: Key<*> = if (baseKey is
        AbstractCoroutineContextKey<*, *>)
        baseKey.topmostKey else baseKey

    internal fun tryCast(element: Element): E? = safeCast(element)
    internal fun isSubKey(key: Key<*>): Boolean = key == this ||
        topmostKey === key

    /**
     * Returns the current element if it is associated with the given [key] in a polymorphic manner
     * or 'null' otherwise.
     */
    @SinceKotlin("1.3")
    fun @ExperimentalStdlibApi public
        get(key: Key<E>): E? =
        if (baseKey == key) getPolymorphicKey() else
        if (this.key == key) this as E else
        public override fun <E : Element>
Element.getPolymorphicElement(key: Key<E>): E? {
    if (key is AbstractCoroutineContextKey<*/*>) {
        return if (key.isSubKey(this.key) && key.tryCast(this) != null)
            EmptyCoroutineContext else this
    } else return if (this.key === key) EmptyCoroutineContext else this
    return if (this.key === key) EmptyCoroutineContext else this
}

/**
 * Returns empty coroutine context if the element is associated with the given [key] in a polymorphic manner or 'null' otherwise.
 * This method returns empty context if either [Element.key] is equal to the given [key] or if the [key] is associated with [Element.key] via [AbstractCoroutineContextKey].
 * See [AbstractCoroutineContextKey] for the example of usage.
 * @since Kotlin 1.3
 * @internal
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Element.minusPolymorphicKey(key: Key<*>): CoroutineContext {
    if (key is AbstractCoroutineContextKey<*/*>) {
        return if (key.isSubKey(this.key) && key.tryCast(this) != null)
            EmptyCoroutineContext else this
    } else return if (this.key === key) EmptyCoroutineContext else this
}

/**
 * An empty coroutine context.
 */
@SinceKotlin("1.3")
public object EmptyCoroutineContext : CoroutineContext, Serializable {
    private const val serialVersionUID: Long = 0
    private fun readResolve(): Any = EmptyCoroutineContext

    public override fun <E : Element> get(key: Key<E>): E? = null
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun plus(context: CoroutineContext): CoroutineContext = context
    public override fun minusKey(key: Key<*>): CoroutineContext {
        return when {
            context === this -> EmptyCoroutineContext
            context === EmptyCoroutineContext -> this
            else ->
                context.minusKey(key)
        }
    }

    public override fun hashCode(): Int = 0
    public override fun toString(): String = "EmptyCoroutineContext"
}

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines.intrinsics

import kotlin.contracts.*
import kotlin.coroutines.*
import kotlin.internal.InlineOnly
and will not return any result immediately. In this case, the [Continuation] provided to the [block] shall be
resumed by invoking [Continuation.resumeWith] at some moment in the *future* when the result becomes
available to resume the computation. *Otherwise, the return value of the [block] must have a type assignable
to [T] and represents the result of this suspend function. *It means that the execution was not suspended and the
[Continuation] provided to the [block] shall not be invoked. *As the result type of the [block] is declared as
`Any?` and cannot be correctly type-checked, its proper return type remains on the conscience of the suspend
function's author. *Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's
thread without going through the [ContinuationInterceptor] that might be present in the coroutine's
[CoroutineContext]. *It is the invoker's responsibility to ensure that a proper invocation context is established.
[Continuation.intercepted] can be used to acquire the intercepted continuation.

Note that it is not
recommended to call either [Continuation.resume] nor [Continuation.resumeWithException] functions
synchronously in the same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way
to obtain current continuation instance.

suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE) } throw NotImplementedError("Implementation of
suspendCoroutineUninterceptedOrReturn is intrinsic")
}

/**
 * This value is used as a return value of [suspendCoroutineUninterceptedOrReturn] `block` argument to state that
 * the execution was suspended and will not return any result immediately. *Note: this value should not be used in general code.**
 * Using it outside of the context of [suspendCoroutineUninterceptedOrReturn] function return value
 * (including, but not limited to, storing this value in other functions, etc) can lead to unspecified
 * behavior of the code.
 * It is implemented as property with getter to avoid ProGuard `<clinit>` problem with
 * multofile IntrinsicsKt class
 */
@SinceKotlin("1.3")
@kotlin.experimental.InlineOnly
public val COROUTINE_SUSPENDED: Any get() =
    CoroutineSingletons.COROUTINE_SUSPENDED
actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin)

Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a
requirement on the language version. This annotation is erased at compile time: its arguments are stored in
a more compact form in the Kotlin metadata.

@Target(AnnotationTarget.CLASS, AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.CONSTRUCTOR,
AnnotationTarget.TYPEALIAS)
@Retention(AnnotationRetention.SOURCE)
@Repeatable
@SinceKotlin("1.2")
internal annotation class RequireKotlin(
    val version: String,
    val message: String = "",
    val level: DeprecationLevel = DeprecationLevel.ERROR,
    val versionKind: RequireKotlinVersionKind = RequireKotlinVersionKind.LANGUAGE_VERSION,
    val errorCode: Int = -1
) * The kind of the version that is required by [RequireKotlin].

API_VERSION

* Specifies that this declaration is a part of special DSL, used for constructing
function's contract.

*Kotlin 1.2*

Calls which are called after the change of the property is made. The value of the property

* has already been changed when this callback is invoked.

* @param onChange the callback which is called before a change to the property value is attempted.

* The value of the property hasn't been changed yet, when this callback is invoked.

* If the callback returns "true" the value of the property is being set to the new value, and if the callback returns "false" the new value is discarded and the property remains its old value.

@sample samples.properties.Delegates.observableDelegate

@sample samples.properties.Delegates.vetoableDelegate

Base interface that can be used for

implementing property delegates of read-only properties.

This is provided only for convenience; you don't have to extend this interface
* as long as your property delegate has methods with the same signatures.
@param T the type of object which owns the delegated property.

@public fun interface ReadOnlyProperty<in T, out V> {
    /**
     * Returns the value of the property for the given object.
     * @param thisRef the object for which the value is requested.
     * @param property the metadata for the property.
     * @return the property value.
     */
    public operator fun getValue(thisRef: T, property: KProperty<*>): V
}

/**
 * Base interface that can be used for implementing property delegates of read-write properties.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your property delegate has methods with the same signatures.
 * @param T the type of object which owns the delegated property.
 * @param V the type of the property value.
 */
public interface ReadWriteProperty<in T, V> : ReadOnlyProperty<T, V> {
    /**
     * Returns the value of the property for the given object.
     * @param thisRef the object for which the value is requested.
     * @param property the metadata for the property.
     * @return the property value.
     */
    public override operator fun getValue(thisRef: T, property: KProperty<*>): V
    /**
     * Sets the value of the property for the given object.
     * @param thisRef the object for which the value is requested.
     * @param property the metadata for the property.
     * @return the property value.
     */
    public override fun setValue(thisRef: T, property: KProperty<*>, value: V)
}

/**
 * Base interface that can be used for implementing property delegate providers.
 *
 * This is provided only for convenience; you don't have to extend this interface as long as your delegate provider has a method with the same signature.
 * @param T the type of object which owns the delegated property.
 * @param D the type of property delegates this provider provides.
 */
@SinceKotlin("1.4")
public fun interface PropertyDelegateProvider<in T, out D> {
    /**
     * Returns the delegate of the property for the given object.
     *
     * This function can be used to extend the logic of creating the object (e.g. perform validation checks)
     * to which the property implementation is delegated.
     * @param thisRef the object for which property delegate is requested.
     * @param property the metadata for the property.
     * @return the property delegate.
     */
    public operator fun provideDelegate(thisRef: T, property: KProperty<*>): D
}

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Implements the core logic of a property delegate for a read/write property that calls callback functions when changed.
 * @param initialValue the initial value of the property.
 */
public abstract class ObservableProperty<V>(initialValue: V) : ReadWriteProperty<Any?, V> {
    private var value = initialValue

    /**
     * The callback which is called before a change to the property value is attempted.
     * The value of the property hasn't been changed yet, when this callback is invoked.
     * If the callback returns `true` the value of the property is being set to the new value,
     * and if the callback returns `false` the new value is discarded and the property remains its old value.
     */
    protected open fun beforeChange(property: KProperty<*>, oldValue: V, newValue: V): Boolean = true

    /**
     * The callback which is called after the change of the property is made.
     */
    protected open fun afterChange(property: KProperty<*>, oldValue: V, newValue: V): Unit {}

    public override fun getValue(thisRef: Any?, property: KProperty<*>): V {
        return value
    }

    public override fun setValue(thisRef: Any?, property: KProperty<*>, value: V) {
        val oldValue = this.value
        if (!beforeChange(property, oldValue, value)) {
            return
        }
        this.value = value
        afterChange(property, oldValue, value)
    }
}
An extension operator that allows delegating a mutable property of type [V] to a property reference to a mutable property of the same type [V].

The reference is without a receiver, i.e. it either references a top-level property or has the receiver bound to it. The reference to a property is a property to a reference to a property of the same type [V].

Example:

```kotlin
// A class Login with a property incorrectAttemptCounter
class Login(val username: String, var incorrectAttemptCounter: Int = 0)
val defaultLogin = Login("Admin")
var defaultLoginAttempts by defaultLogin.incorrectAttemptCounter
set(value) {
    defaultLogin.incorrectAttemptCounter = value
}
```

An extension operator that allows delegating a read-only member or extension property of type [V] to a property reference to a read-only member or extension property of the same type [V].

Example:

```kotlin
// A class Login with a property incorrectAttemptCounter
class Login(val username: String)
val Login.user by Login::username
```

An extension operator that allows delegating a mutable member or extension property of type [V] to a property reference to a mutable member or extension property of the same type [V].

Example:

```kotlin
// A class Login with a property incorrectAttemptCounter
class Login(val username: String, var incorrectAttemptCounter: Int)
var Login.attempts by Login::incorrectAttemptCounter
```

An abstract class that is implemented by random number generator algorithms.

The companion object Random.Default is the default instance of Random.

To get a seeded instance of random generator use Random function.

```kotlin
class Random {
    fun nextBits(bitCount: Int): Int
    fun nextInt(): Int
    fun nextInt(until: Int): Int
    fun nextInt(from: Int, until: Int): Int
}
```

The behavior is unspecified if the `until` boundary is not within the range of the generator's range.

```kotlin
class Random {
    fun nextBits(bitCount: Int): Int
    fun nextInt(): Int
    fun nextInt(until: Int): Int
    fun nextInt(from: Int, until: Int): Int
}
```

Throws IllegalArgumentException if `from` is greater than or equal to `until`.

```kotlin
class Random {
    fun nextBits(bitCount: Int): Int
    fun nextInt(): Int
    fun nextInt(until: Int): Int
    fun nextInt(from: Int, until: Int): Int
}
```

## Random.Default

```kotlin
object Random.Default :Random {
    override fun nextBits(bitCount: Int): Int
    override fun nextInt(): Int
    override fun nextInt(until: Int): Int
    override fun nextInt(from: Int, until: Int): Int
}
```

Throws IllegalArgumentException if `from` is greater than or equal to `until`.

```kotlin
class Random {
    fun nextBits(bitCount: Int): Int
    fun nextInt(): Int
    fun nextInt(until: Int): Int
    fun nextInt(from: Int, until: Int): Int
}
```

Throws IllegalArgumentException if `from` is greater than or equal to `until`.

```kotlin
class Random {
    fun nextBits(bitCount: Int): Int
    fun nextInt(): Int
    fun nextInt(until: Int): Int
    fun nextInt(from: Int, until: Int): Int
}
```
Fills the specified byte [array] with random bytes and returns it.

* @return [array] filled with random bytes.

/**
 * Creates a byte array of the specified [size], filled with random bytes.
 */
public fun nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))

/**
 * The default random number generator.
 */
private val defaultRandom: Random = defaultPlatformRandom()

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Int` value.
 * Two generators with the same seed produce the same sequence of values within the same version of Kotlin runtime.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return a sequence of values different from the current one for a given seed.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * Future versions of Kotlin may change the algorithm of this seeded number generator so that it will return a sequence of values different from the current one for a given seed.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * On JVM the returned generator is NOT thread-safe. Do not invoke it from multiple threads without proper synchronization.
 * @throws IllegalArgumentException if [range] is empty.
 * Gets the next random `Int` value uniformly distributed in the specified [range]:
 * `range.start` inclusive to `range.endInclusive` inclusive.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
 * @throws IllegalArgumentException if [range] is empty.
bitCount).shr(31)

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Double, until: Double) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun boundsErrorMessage(from: Any, until: Any) = "Random range is empty: [$from, $until)."

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be
 * found in the license/LICENSE.txt file.*/

package kotlin.random

/**
 * Gets the next random [UInt] from the random number generator.
 * * Generates a [UInt] random value uniformly distributed between
 * [UInt.MIN_VALUE] and [UInt.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt = nextInt().toUInt()

/**
 * Gets the next random [UInt] from the random number generator less than the specified
 * [until] bound.
 * * Generates a [UInt] random value uniformly distributed between '0' (inclusive) and the
 * specified [until] bound.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(until: UInt): UInt = nextUInt(0u, until)

/**
 * Gets the next random [UInt] from the random number generator in the specified
 * range.
 * * Generates a [UInt] random value uniformly distributed in the specified
 * range: from `range.start` inclusive to `range.endInclusive` inclusive.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(range: UIntRange): UInt = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < UInt.MAX_VALUE -> nextUInt(range.first, range.last + 1u)
    range.first > UInt.MIN_VALUE -> nextUInt(range.first - 1u, range.last) + 1u
    else -> nextUInt()
}

/**
 * Gets the next random [ULong] from the random number generator.
 * * Generates a [ULong] random value uniformly distributed between
 * [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(): ULong = nextLong().toULong()

/**
 * Gets the next random [ULong] from the random number generator less than the specified
 * [until] bound.
 * * Generates a [ULong] random value uniformly distributed between '0' (inclusive) and the
 * specified [until] bound.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(until: ULong): ULong = nextULong(0uL, until)

/**
 * Gets the next random [ULong] from the random number generator in the specified range.
 * * Generates a [ULong] random value uniformly distributed between the specified
 * [from] (inclusive) and [until] (exclusive) bounds.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(from: ULong, until: ULong): ULong = when {
    from > until -> throw IllegalArgumentException("Cannot get random in empty range: $from...$until")
    else -> nextULong()
ULong.MIN_VALUE -> nextULong(range.first - 1u, range.last) + 1u
else -> nextULong(range.first - 1u, range.last) + 1u

/**
 * Fills the specified unsigned byte [array] with random bytes and returns it.
 * @return [array] filled with random bytes.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(range: ClosedRange<ULong>): UByteArray {
    @ExperimentalUnsignedTypes
    @IntRange(minValue = ULong.MIN_VALUE, maxValue = ULong.MAX_VALUE)
    override fun nextULong(): ULong = nextBytes(range.first - 1u, range.last + 1u)
    else -> nextULong()
}

/**
 * Fills
 * the specified unsigned byte [array] with random bytes and returns it.
 * @return [array] filled with random bytes.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextBytes(array: UByteArray): UByteArray
    return array
    @ExperimentalUnsignedTypes
    @IntArray(minValue = ULong.MIN_VALUE, maxValue = ULong.MAX_VALUE)
    override fun nextBytes(range: ClosedRange<ULong>): UByteArray
    @IntRange(minValue = ULong.MIN_VALUE, maxValue = ULong.MAX_VALUE)
    return nextULong(range.first - 1u, range.last + 1u)
    else -> nextULong()

/**
 * Creates an unsigned byte array of the specified [size], filled with random bytes.
 * @param size [size] to fill.
 * @return [array] with the specified [size].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextBytes(size: Int): UByteArray
    return nextBytes(size).asUByteArray()

/**
 * Fills a subrange of the specified `UByte` [array] starting from [fromIndex]
 * inclusive and ending to [toIndex] exclusive with random UBytes.
 * @param array [array] to fill.
 * @param fromIndex [fromIndex] to start filling.
 * @param toIndex [toIndex] to end filling.
 * @return [array] with the specified [fromIndex] to [toIndex] filled with random bytes.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUBytes(array: UByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray
    return array

@ExperimentalUnsignedTypes
internal fun checkUIntRangeBounds(from: UInt, until: UInt) = require(until > from) { boundsErrorMessage(from, until) }
@ExperimentalUnsignedTypes
internal fun checkULongRangeBounds(from: ULong, until: ULong) = require(until > from) { boundsErrorMessage(from, until) }

package kotlin.random

/**
 * Random number generator, using Marsaglia's "xorwow" algorithm
 * Cycles after 2^192 - 2^32 repetitions.
 * For more details, see Marsaglia, George (July 2003). "Xorshift RNGs". Journal of Statistical Software. 8 (14). doi:10.18637/jss.v008.i14
 * Available at https://www.jstatsoft.org/v08/i14/paper
 */
internal class XorWowRandom
internal constructor(
    private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private var addend: Int
) : Random() {

    internal constructor(seed1: Int, seed2: Int) :
            this(seed1, seed2, 0, 0,
    seed1.inv(), (seed1 shl 10) xor (seed2 ushr 4))

    init {
        require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero element." }
        // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64
        repeat(64) { nextInt() }
    }

    override fun nextInt(): Int {
        // Equivalent to the xorxow algorithm
        var t = x
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        v = t
        addend += 362437
        return t + addend
    }

    override
    fun nextBits(bitCount: Int): Int =
        nextInt().takeUpperBits(bitCount)
}

package kotlin.ranges

/**
 * Represents a range of [Comparable] values.
 */
private open class ComparableRange<T : Comparable<T>>(
    override val start: T,
    override val endInclusive: T
) : ClosedRange<T> {

    override
    fun equals(other: Any?): Boolean {
        return other is ComparableRange<*> && (isEmpty() && other.isEmpty() ||
                start == other.start && endInclusive == other.endInclusive)
    }

    override
    fun hashCode(): Int =
        if (isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()

    override
    fun toString(): String = "$start..$endInclusive"
}

/**
 * Creates a range from this [Comparable] value to the specified [that] value.
 * This value needs to be smaller than [that] value, otherwise the returned range will be empty.
 * @sample samples.ranges.Ranges.rangeFromComparable
 */
public operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)

/**
 * Represents a range of floating point numbers.
 * Extends [ClosedRange] interface providing custom operation
 * [contains] for comparing values of range domain type.
 */
@SinceKotlin("1.1")
public interface ClosedFloatingPointRange<T : Comparable<T>> : ClosedRange<T> {
    override
    fun contains(value: T): Boolean = lessThanOrEquals(start, value) && lessThanOrEquals(value, endInclusive)
    override
    fun isEmpty(): Boolean = false

    /**
     * Compares two values of range domain type and returns true
     */
    fun lessThanOrEquals(value: T): Boolean = value <= start && value >= endInclusive
}

Open Source Used In DNAC Platform Arica 9990
if first is less than or equal to second.

/**
 * Copyright 2010-2019
 * Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("KClasses")
@file:Suppress("UNCHECKED_CAST")

package kotlin.reflect

import kotlin.internal.LowPriorityInOverloadResolution

/**
 * Casts the given `value` to the class represented by this `KClass` object.
 * Returns `null` if the value is `null` or if it is not an instance of this class.
 * This is an experimental function that behaves as a similar function from kotlin.reflect.full.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@LowPriorityInOverloadResolution
fun <T : Any> KClass<T>.cast(value: Any?): T {
    if (!isInstance(value)) throw ClassCastException("Value cannot be cast to $qualifiedOrSimpleName")
    return value as T
}

// TODO: replace with qualifiedName when it is fully supported in K/JS
internal expect val KClass<*>.qualifiedOrSimpleName: String?

/**
 * Casts the given `value` to the class represented by this `KClass` object.
 * Returns `null` if the value is `null` or if it is not an instance of this class.
 * This is an experimental function that behaves as a similar function from kotlin.reflect.full.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@LowPriorityInOverloadResolution
fun <T : Any> KClass<T>.safeCast(value: Any?): T? {
    return if (isInstance(value)) value as T else
null
}"

Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* Represents a type projection. Type projection is usually the argument to another type in a type usage.

* Represents the KTypeProjection

* Represents the use-site variance specified in the projection, or ‘null’ if this is a star projection.

* Represents the type specified in the projection, or ‘null’ if this is a star projection.

* Represents the initial type requires(variance == null) == (type == null)

* Represents the projection variance $variance requires type to be specified.

* Represents the public val type: KType

* Represents the @PublishedApi

* Represents the internal val star: KTypeProjection =

* Represents the invariant type.

* Represents the contravariant type.

* Represents the covariant type.

* Represents the runtime representation of the given reified type [T] as an instance of [KType].

* Represents the UnsupportedOperationException("This function is implemented as an intrinsic on all supported platforms.")

* Represents the API public inline fun <T> typeOf(): KType

* Represents the @package kotlin.reflect

* Represents the @JvmStatic

* Represents the @PublishedApi

* Represents the internal val star: KTypeProjection =

* Represents the invariant type.

* Represents the contravariant type.

* Represents the covariant type.

* Represents the runtime representation of the given reified type [T] as an instance of [KType].

* Represents the UnsupportedOperationException("This function is implemented as an intrinsic on all supported platforms.")

* Represents the API public inline fun <T> typeOf(): KType

* Represents the @package kotlin.reflect

* Represents the @JvmStatic

* Represents the @PublishedApi

* Represents the internal val star: KTypeProjection =

* Represents the invariant type.

* Represents the contravariant type.

* Represents the covariant type.

* Represents the runtime representation of the given reified type [T] as an instance of [KType].
package kotlin.text

/**
 * An object to which char sequences and values can be appended.
 */
expect interface Appendable {
    /**
     * Appends the specified character [value] to this Appendable and returns this instance.
    */
    fun append(value: Char): Appendable

    /**
     * Appends the specified character sequence [value] to this Appendable and returns this instance.
    */
    fun append(value: CharSequence?): Appendable

    /**
     * Appends a subsequence of the specified character sequence [value] to this Appendable and returns this instance.
    */
    fun append(value: CharSequence?, startIndex: Int, endIndex: Int): Appendable
}

public fun <T : Appendable> T.appendRange(value: CharSequence, startIndex: Int, endIndex: Int): T {
    return append(value, startIndex, endIndex) as T
}

/**
 * Appends all arguments to the given Appendable.
 */
public fun <T : Appendable> T.append(vararg value: CharSequence?): T {
    for (item in value)
        append(item)
    return this
}

/** Appends a line feed character (`\n`) to this Appendable. */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Appendable.appendLine(): Appendable = append('\n')

/** Appends value to the given Appendable and a line feed character (`\n`) after it. */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Appendable.appendLine(value: CharSequence?): Appendable = append(value).appendLine()

/** Appends value to the given Appendable and a line feed character (`\n`) after it. */
@SinceKotlin("1.4")
@kotlin.internal.InlineOnly
public inline fun Appendable.appendLine(value: Char):


internal fun <T> Appendable.appendElement(element: T,
    transform: ((T) -> CharSequence)?) {
    when {
        transform != null -> append(transform(element))
        element is CharSequence? -> append(element)
        element is Char -> append(element)
        else ->
            append(element.toString())
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
Char.isSurrogate(): Boolean = this in Char.MIN_SURROGATE..Char.MAX_SURROGATE

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKit")

package kotlin.text

/**
 * Trims leading whitespace characters followed by [marginPrefix] from every line of a source string
 * and removes the first and the last lines if they are blank (notice difference blank vs empty).
 * Doesn't affect a line if it doesn't contain [marginPrefix] except the first and the last blank lines.
 * Doesn't preserve the original line endings.
 * @param marginPrefix non-blank string, which is used as a margin delimiter.
 * Default is `\` (pipe character).
 * @sample samples.text.Strings.trimMargin
 * @see trimIndent
 */
public fun String.trimMargin(marginPrefix: String = `\`): String =
    replaceIndentByMargin("", marginPrefix)

/**
 * Detects indent by [marginPrefix] as it does [trimMargin] and replace it with [newIndent].
 * @param marginPrefix non-blank string, which is used as a margin delimiter.
 * Default is `\` (pipe character).
 * @sample samples.text.Strings.trimIndent
 * @see trimMargin
 */
public fun String.trimIndent(): String = replaceIndent("")

/**
 * Detects a common minimal indent of all the input lines, removes it from every line and also removes the first and the last non-blank lines if they are blank (notice difference blank vs empty).
 * @param marginPrefix non-blank string, which is used as a margin delimiter.
 * @sample samples.text.Strings.trimIndent
 */
public fun String.replaceIndent(newIndent: String = ""): String {
    val lines = lines()

    return lines.reindent(length + newIndent.length * lines.size,
        getIndentFunction(newIndent), { line ->
            val firstNonWhitespaceIndex = line.indexOfFirst { !it.isWhitespace() }

            when {
                firstNonWhitespaceIndex == -1 -> null
                line.startsWith(marginPrefix, firstNonWhitespaceIndex) -> line.substring(firstNonWhitespaceIndex)
                else -> null
            }
        })
}

/**
 * Prepends [indent] to every line of the original string.
 * @sample samples.text.Strings.prependIndent
 */
public fun String.prependIndent(indent: String = "    "): String =
    lineSequence()
        .map {
            when {
                it.isBlank() -> {
                    when {
                        it.length < indent.length -> indent
                        else -> it
                    }
                }
                else -> indent + it
            }
        }
        .joinToString("\n")

private fun String.indentWidth(): Int = indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }

private fun getIndentFunction(indent: String) = when {
    indent.isEmpty() -> { line: String -> line }
    else -> { line: String -> indent + line }
}

private inline fun List<String>.reindent(
    resultSizeEstimate: Int,
    indentAddFunction: (String) -> String,
    indentCutFunction: (String) -> String?
): String {
    val lastIndex = lastIndex
    return
        mapIndexedNotNull { index, value ->
            if ((index == 0 || index == lastIndex) && value.isBlank())
                null
            else
                indentCutFunction(value)?.let(indentAddFunction) ?: value
        }
    .joinTo(StringBuilder(resultSizeEstimate), "\n")
    .toString()
}

package kotlin.text

/**
 * Defines names for Unicode symbols used in proper Typography.
 */
public object Typography {
    /** The character " \u2013 quotation mark */
    public const val quote: Char = '\u0022'
    /** The character $ \u2013 dollar sign */
    public const val dollar: Char = '\u0024'
    /** The character & \u2013 ampersand */
    public const val amp: Char = '\u0026'
    /** The character < \u2013 less-than sign */
    public const val less: Char = '\u003C'
    /** The character > \u2013 greater-than sign */
    public const val greater: Char = '\u003E'
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The non-
### Open Source Used In DNAC Platform

Arica

```kotlin
// Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.text

/**
 * Represents a collection of captured groups in a single match of a regular expression.
 * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
 * An element of the collection at the particular index can be `null`, if the corresponding group in the regular expression is optional
 * and there was no match captured by that group.
 * @param groupCount The count of groups in the regular expression.
 * @param index The index of the group.
 * @return A collection of groups matched by the regular expression.
 * Groups are indexed from 1 to the count of groups in the regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
 */
public interface MatchGroupCollection : Collection<MatchGroup?> {

    /** Returns a group with the specified [index].
     * @param index The index of the group.
     * @return An instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.
     * @throws IllegalArgumentException if there is no group with the specified [index] defined in the regex pattern.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public operator fun get(index: Int): MatchGroup?

    /**
     * A list of matched indexed group values.
     * This list has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     */
    public val indexedGroups: List<MatchGroup>
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /** The range of indices in the original string where match was captured. */
    public val range: IntRange
    /** The substring from the input string captured by this match. */
    public val value: String
    /** A list of matched indexed group values. */
    public val indexedGroups: List<MatchGroup>
}
```

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * *
fun isNegative(): Boolean = value < 0
    /** Returns true, if the duration value is greater than zero. */
    public fun isPositive(): Boolean = value > 0
    /** Returns true, if the duration value is infinite. */
    public fun isInfinite(): Boolean = value.isInfinite()
    /** Returns true, if the duration value is finite. */
    public fun isFinite(): Boolean = value.isFinite()
    /** Returns the absolute value of this value. The result is always non-negative. */
    public val absoluteValue: Duration get() = if (isNegative()) -this else this
    override fun isFinite(): Boolean = value.isFinite()
    /** Returns a duration whose value is the sum of this and [other] duration values. */
    public operator fun plus(other: Duration): Duration = Duration(value + other.value)
    /** Returns a duration whose value is the difference between this and [other] duration values. */
    public operator fun minus(other: Duration): Duration = Duration(value - other.value)
    /** Returns a duration whose value is less than zero. */
    public operator fun times(scale: Int): Duration = Duration(value * scale)
    /** Returns a duration whose value is greater than zero. */
    public operator fun times(scale: Double): Duration = Duration(value * scale)
    /** Returns a duration whose value is equal to zero. */
    public operator fun times(scale: Double): Duration = Duration(0)
    /** Returns a duration whose value is less than zero. */
    public operator fun times(scale: Int): Duration = Duration(value / scale)
    /** Returns a number that is the ratio of this and [other] duration values. */
    public operator fun div(scale: Double): Duration = Duration(value / scale)
    /** Returns a duration whose value is the difference between this and other duration values. */
    public operator fun div(other: Duration): Duration = Duration(value / other.value)
    /** Returns true, if the duration value is less than zero. */
    public fun isNegative(): Boolean = value < 0
    /** Returns true, if the duration value is greater than zero. */
    public fun isPositive(): Boolean = value > 0
    /** Returns true, if the duration value is infinite. */
    public fun isInfinite(): Boolean = value.isInfinite()
    /** Returns true, if the duration value is finite. */
    public fun isFinite(): Boolean = value.isFinite()
    /** Returns the absolute value of this value. The result is always non-negative. */
    public val absoluteValue: Duration get() = if (isNegative()) -this else this
    override fun isFinite(): Boolean = value.isFinite()
    /** Returns a duration whose value is the sum of this and [other] duration values. */
    public operator fun plus(other: Duration): Duration = Duration(value + other.value)
    /** Returns a duration whose value is the difference between this and [other] duration values. */
    public operator fun minus(other: Duration): Duration = Duration(value - other.value)
    /** Returns a duration whose value is less than zero. */
    public operator fun times(scale: Int): Duration = Duration(value * scale)
    /** Returns a duration whose value is greater than zero. */
    public operator fun times(scale: Double): Duration = Duration(value * scale)
    /** Returns a duration whose value is equal to zero. */
    public operator fun times(scale: Double): Duration = Duration(0)
    /** Returns a duration whose value is less than zero. */
    public operator fun times(scale: Int): Duration = Duration(value / scale)
    /** Returns a number that is the ratio of this and [other] duration values. */
    public operator fun div(scale: Double): Duration = Duration(value / scale)
    /** Returns a duration whose value is the difference between this and other duration values. */
    public operator fun div(other: Duration): Duration = Duration(value / other.value)
    /** Returns true, if the duration value is less than zero. */
    public fun isNegative(): Boolean = value < 0
    /** Returns true, if the duration value is greater than zero. */
    public fun isPositive(): Boolean = value > 0
    /** Returns true, if the duration value is infinite. */
    public fun isInfinite(): Boolean = value.isInfinite()
    /** Returns true, if the duration value is finite. */
    public fun isFinite(): Boolean = value.isFinite()
    /** Returns the absolute value of this value. The result is always non-negative. */
    public val absoluteValue: Duration get() = if (isNegative()) -this else this
    override fun isFinite(): Boolean = value.isFinite()
secondsComponent, nanosecondsComponent) /* Splits this duration into seconds, and
nanoseconds and executes the given [action] with these components. */ The result of [action] is returned as the
result of this function. * 'nanoseconds' represents the whole number of nanoseconds in this duration,
and its absolute value is less than 1_000_000_000; * 'seconds' represents the whole number of seconds in this
duration. * If the value doesn't fit in [Long] range, i.e. it's greater than [Long.MAX_VALUE] or less than
[Long.MIN_VALUE]. * it is coerced into that range. */ public inline fun <T> toComponents(action: (seconds: Long, nanosecondsComponent: Int) -> T): T { contract { callsInPlace(action, InvocationKind.EXACTLY_ONCE) } return action(inSeconds.toLong(), nanosecondsComponent) }

internal val hoursComponent: Int get() = (inHours % 24).toInt()
internal val minutesComponent: Int get() = (inMinutes % 60).toInt()
internal val secondsComponent: Int get() = (inSeconds % 60).toInt()
internal val nanosecondsComponent: Int get() = (inNanoseconds % 1e9).toInt()

// conversion to units

public val inDays: Double get() = toDouble(DurationUnit.DAYS)
public val inHours: Double get() = toDouble(DurationUnit.HOURS)
public val inMinutes: Double get() = toDouble(DurationUnit.MINUTES)
public val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)
public val inMilliseconds: Double get() = toDouble(DurationUnit.MILLISECONDS)
public val inMicroseconds: Double get() = toDouble(DurationUnit.MICROSECONDS)
public val inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS)

/** The value of this duration expressed as a [Double] number of nanoseconds. */
public fun toLongNanoseconds(): Long = toLong(DurationUnit.NANOSECONDS)

/** The value of this duration expressed as a [Double] number of milliseconds. */
public fun toLongMilliseconds(): Long = toLong(DurationUnit.MILLISECONDS)

/** The value of this duration expressed as a [Double] number of days. */

/** The value of this duration expressed as a [Double] number of minutes. */

/** The value of this duration expressed as a [Double] number of seconds. */
public fun toString(): String = when {
    isInfinite() -> value.toString()
    value == 0.0 -> "0s"
    else -> {
        var scientific = false
        var maxDecimals = 0
        val unit = when {
            val absNs = absoluteValue.inNanoseconds
            absNs < 1e-15 -> {
                val value = 0.0 -> "0s"
                else -> {
                    val absNs = absoluteValue.inNanoseconds
                    var scientific = false
                    var maxDecimals = 0
                    val unit = when {
                        
                    }
                    
                }
            }
        }
    }
}
maxDecimals = 7

val absNs < 1e3 -> DurationUnit.NANOSECONDS
val absNs < 1e9 -> DurationUnit.MILLISECONECONDS
val absNs < 1000e9 -> DurationUnit.SECONDSC
val absNs < 60_000e9 -> DurationUnit.MINUTES
val absNs < 3600_000e9 -> DurationUnit.HOURS
val absNs < 86400e9 -> DurationUnit.DAYS
val else -> DurationUnit.DAYS.also {
    scientific = true
} val value = toDouble(unit) when {
    scientific -> formatScientific(value)
    maxDecimals > 0 -> formatUpToDecimals(value, maxDecimals)
    else -> formatToExactDecimals(value, precision(abs(value)))
} + unit.shortName()

val value = toDouble(unit)

when {
    scientific -> formatScientific(value)
    maxDecimals > 0 -> formatUpToDecimals(value, maxDecimals)
    else -> formatToExactDecimals(value, precision(abs(value)))
} + unit.shortName()

private fun precision(value: Double): Int = when {
    value < 1 -> 3
    value < 10 -> 2
    value < 100 -> 1
    else -> 0
}

/**
 * Returns a string representation of this duration value expressed in the given [unit] and formatted with the specified [decimals] number of digits after decimal point.
 * The returned value is formatted as `"Infinity"` without unit.
 * @returns a string representation of this duration value expressed in the given [unit] and formatted with the specified [decimals] number of digits after decimal point.
 * @throws IllegalArgumentException if [decimals] is less than zero.
 * @sample samples.time.Durations.toStringDecimals
 */
public fun toString(unit: DurationUnit, decimals: Int = 0): String {
    require(decimals >= 0) { "decimals must be not negative, but was $decimals" }
    if (isInfinite()) return value.toString()
    val number = toDouble(unit)
    return when {
        abs(number) < 1e14 -> formatToExactDecimals(number, decimals.coerceAtMost(12))
        else -> formatScientific(number) + unit.shortName()
    }

/**
 * Returns an ISO-8601 based string representation of this duration.
 * The returned value is presented in the format `PThHmMs.fS`, where `h`, `m`, `s` are the integer components of this duration (see [toComponents]) and `f` is a fractional part of second. Depending on the roundness of the value the fractional part can be formatted with either `0`, `3`, or `9` decimal digits.
 * @returns an ISO-8601 based string representation of this duration.
 * @throws IllegalArgumentException if [decimals] is less than zero.
 * @sample samples.time.Durations.toIsoString
 */
@OptIn(ExperimentalStdlibApi::class)
public fun toIsoString(): String = buildString {
    if (isNegative()) append('-')
    append("PT")
    absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->
        val hasHours = hours != 0
        val hasSeconds = seconds != 0 || nanoseconds != 0
        val hasMinutes = minutes != 0 || (hasSeconds && hasHours)
        if (hasHours) {
            append(hours).append('H')
        }
        if (hasMinutes) {
            append(minutes).append('M')
        }
        if (hasSeconds || (!hasHours && !hasMinutes)) {
            append(seconds)
                if (nanoseconds != 0) {
                    append('.')
                    val nss = nanoseconds.toString().padStart(9, '0')
                    when {
                        nanoseconds % 1_000_000 == 0 -> appendRange(nss, 0, 3)
                        nanoseconds % 1_000 == 0 -> appendRange(nss, 0, 6)
                        else -> appendRange(nss, 0, 9)
                    }
                }
                append('S')
            }
        }
    }

/**
 * Returns a [Duration] equal to this [Int] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/**
 * Returns a [Duration] equal to this [Long] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/**
 * Returns a [Duration] equal to this [Double] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

// constructing from number of units
// extension functions
/**
 * Returns a [Duration] equal to this [Int] number of the specified [unit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Int.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/**
 * Returns a [Duration] equal to this [Long] number of the specified [unit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Long.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/**
 * Returns a [Duration] equal to this [Double] number of the specified [unit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Double.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

// constructing from number of units
// extension properties
/**
 * Returns a [Duration] equal to this [Int] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
val Int.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/**
 * Returns a [Duration] equal to this [Long] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/**
 * Returns a [Duration] equal to this [Double] number of nanoseconds.
 */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)
/** Returns a [Duration] equal to this [Int] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Long] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Double] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Int] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
val Int.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Long] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.milliseconds get() =
    toDuration(DurationUnit.MILLISECOS)

/** Returns a [Duration] equal to this [Double] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Int] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Long] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Double] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Int] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Long] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Double] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.minutes get() =
    toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Int] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.hours get() =
    toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Long] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.hours get() =
    toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Double] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.hours get() =
    toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Int] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Long] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
val Long.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Double] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double.days get() =
    toDuration(DurationUnit.DAYS)

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
val Int@kotlin.internal.InlineOnly
    public inline operator fun Int.times(duration: Duration): Duration =
        duration * this

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
val Double@kotlin.internal.InlineOnly
    public inline operator fun Double.times(duration: Duration):
        Duration = duration * this

internal expect fun formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double, decimals: Int): String
internal expect fun formatScientific(value: Double): String

-open source used in DNAC Platform Arica- 10000
public expect enum class DurationUnit {
    NANOSECONDS,
    MICROSECONDS,
    MILLISECONDS,
    SECONDS,
    MINUTES,
    HOURS,
    DAYS;
}

internal expect fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double

internal fun DurationUnit.shortName(): String = when (this) {
    DurationUnit.NANOSECONDS -> "ns"
    DurationUnit.MICROSECONDS -> "μs"
    DurationUnit.MILLISECONDS -> "ms"
    DurationUnit.SECONDS -> "s"
    DurationUnit.MINUTES -> "m"
    DurationUnit.HOURS -> "h"
    DurationUnit.DAYS -> "d"
}
operator fun plus(duration: Duration): TimeMark = AdjustedTimeMark(this, duration) /* Returns a time mark on the same time source that is behind this time mark by the specified [duration]. */

* The returned time mark is more _early_ when the [duration] is positive, and more _late_ when the [duration] is negative.

* public open operator fun minus(duration: Duration): TimeMark = plus(-duration) /* Returns true if this time mark has passed according to the time source from which this mark was taken. */

* Note that the value returned by this function can change on subsequent invocations.

* If the time source is monotonic, it can change only from 'false' to 'true', namely, when the time mark becomes behind the current point of the time source.

* public fun hasPassedNow(): Boolean = elapsedNow().isNegative() /* Returns false if this time mark has not passed according to the time source from which this mark was taken. */

* Note that the value returned by this function can change on subsequent invocations.

* If the time source is monotonic, it can change only from 'true' to 'false', namely, when the time mark becomes behind the current point of the time source.

* public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative() /* Use TimeSource interface instead. */

private class AdjustedTimeMark(val mark: TimeMark, val adjustment: Duration) : TimeMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): TimeMark = AdjustedTimeMark(mark, adjustment + duration)
}

private class LongTimeMark(private val startedAt: Long, private val timeSource: AbstractLongTimeSource, private val offset: Duration) : TimeMark() {
        override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
        override fun plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)
    }
DoubleTimeMark(private val startedAt: Double, private val timeSource: AbstractDoubleTimeSource, private val offset: Duration) : TimeMark() {
    override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
    override fun plus(duration: Duration): TimeMark = DoubleTimeMark(startedAt, timeSource, offset + duration)

    override fun markNow(): TimeMark = DoubleTimeMark(read(), this, Duration.ZERO)
}

/**
 * A time source that has programmatically updatable readings. It is useful as a predictable source of time in tests.
 * The current reading value can be advanced by the specified duration amount with the operator [plusAssign]:
 * val timeSource = TestTimeSource()
 * timeSource += 10.seconds
 */
@SinceKotlin("1.3")
@ExperimentalTime
public class TestTimeSource : AbstractLongTimeSource(unit = DurationUnit.NANOSECONDS) {
    private var reading: Long = 0L

    override fun read(): Long = reading

    /**
     * Advances the current reading value of this time source by the specified [duration].
     * For example, if the duration being added is '0.6.nanoseconds', the reading doesn't advance because the duration value is rounded to zero nanoseconds.
     */
    public operator fun plusAssign(duration: Duration) {
        val delta = duration.toDouble(unit)
        val longDelta = delta.toLong()
        reading = if (longDelta != Long.MIN_VALUE && longDelta != Long.MAX_VALUE) {
            // when delta fits in long, add it as long
            val newReading = reading + longDelta
            if (reading xor longDelta >= 0 && reading xor newReading < 0)
                throw IllegalStateException("TestTimeSource will overflow if its reading $readingns is advanced by $duration.")
            newReading
        } else {
            // when delta is greater than long, add it as double
            val newReading = reading + delta
            if (newReading > Long.MAX_VALUE || newReading < Long.MIN_VALUE) throw IllegalStateException("TestTimeSource will overflow if its reading $readingns is advanced by $duration.")
            newReading.toLong()
        }
    }

    private fun

    /**
     * Data class representing a result of executing
     */
    inline fun measureTime(block: () -> Unit): Duration {
        contract {
            callsInPlace(block, InvocationKind.EXACTLY_ONCE)
        }
        return TimeSource.Monotonic.measureTime(block)
an action, along with the duration of elapsed time interval.

```kotlin
@SinceKotlin("1.3")@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class, containing both
 * the result of the function execution and the duration of elapsed time interval.
 * The elapsed time is measured with [TimeSource.Monotonic].
 * @property value the result of the action.
 * @property duration the time elapsed to execute the action.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class, containing both
 * the result of the function execution and the duration of elapsed time interval.
 * The elapsed time is measured with [TimeSource.Monotonic].
 * @property value the result of the action.
 * @property duration the time elapsed to execute the action.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given [block] and returns an instance of [TimedValue] class, containing both
 * the result of the function execution and the duration of elapsed time interval.
 * The elapsed time is measured with the specified [TimeSource] instance.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Initiates a call to this deep recursive function using its [invoke] function.
 * @param [T] the function parameter type.
 * @param [R] the function result type.
 * @param block the function body.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public class DeepRecursiveFunction<T, R>(
    internal val block: suspend DeepRecursiveScope<T, R>.(T) -> R
)

/**
 * A scope
class for [DeepRecursiveFunction] function declaration that defines [callRecursive] methods to recursively call this function or another [DeepRecursiveFunction] putting the call activation frame on the heap.

*public abstract suspend fun callRecursive(value: T): R


@RestrictsSuspension
@SinceKotlin("1.4")
@ExperimentalStdlibApi
public sealed class DeepRecursiveScope<T, R> {

    /**
     * Makes recursive call to this [DeepRecursiveFunction] function putting the call activation frame on the heap, as opposed to the actual call stack that is used by a regular recursive call.
     */
    public abstract suspend fun callRecursive(value: T): R

    /**
     * Makes call to the specified [DeepRecursiveFunction] function putting the call activation frame on the heap, as opposed to the actual call stack.
     */

    @Deprecated(
        level = DeprecationLevel.ERROR,
        message = "'invoke' should not be called from DeepRecursiveScope. Use 'callRecursive' to do recursion in the heap instead of the call stack."
    )

    @Suppress("UNUSED_PARAMETER")
    public operator fun DeepRecursiveFunction<*, *>.invoke(value: Any?): Nothing =
        throw UnsupportedOperationException("Should not be called from DeepRecursiveScope")
}

// ================== Implementation

@ExperimentalStdlibApi
private typealias DeepRecursiveFunctionBlock = suspend DeepRecursiveScope<*, *>.(Any?) -> Any?

private val UNDEFINED_RESULT = Result.success(COROUTINE_SUSPENDED)

@Suppress("UNCHECKED_CAST")
@ExperimentalStdlibApi
private class DeepRecursiveScopeImpl<T, R>(
    block: suspend DeepRecursiveScope<T, R>.(T) -> R,
    value: T
) : DeepRecursiveScope<T, R>(), Continuation<R> {

    // Active function block
    private var function: DeepRecursiveFunctionBlock = block as DeepRecursiveFunctionBlock

    // Value to call function with
    private var value: Any? = value

    // Continuation of the current call
    private var cont: Continuation<Any?>? = this as Continuation<Any?>

    // Completion result (completion of the whole call stack)
    private var result: Result<Any?> = UNDEFINED_RESULT

    override val context: CoroutineContext
        get() = EmptyCoroutineContext

    override fun resumeWith(result: Result<R>) {
        this.cont = null
        this.result = result
    }

    override suspend fun callRecursive(value: T): R =
        suspendCoroutineUninterceptedOrReturn { cont ->
            // calling the same function that is currently active
            this.cont = cont as Continuation<Any?>
            this.value = value
            COROUTINE_SUSPENDED
        }

        suspendCoroutineUninterceptedOrReturn { cont ->
            // calling another recursive function
            val function = block as DeepRecursiveFunctionBlock
            with(this@DeepRecursiveScopeImpl) {
                val currentFunction = this.function
                if (function !== currentFunction) {
                    // calling a different function -- create a trampoline to restore function ref
                    this.function = function
                    this.cont = crossFunctionCompletion(currentFunction, cont)
                } else {
                    // calling the same function -- direct
                    this.cont = cont as Continuation<Any?>
                    this.value = value
                }
                COROUTINE_SUSPENDED
            }

            //Restore to the trampoline
            this.cont = cont
            this.result = result
            this.function = function
            this.value = value
        }

    private fun crossFunctionCompletion(
        currentFunction: DeepRecursiveFunctionBlock,
        cont: Continuation<Any?>
    ): Continuation<Any?> =
        Continuation(EmptyCoroutineContext) {
            this.function = currentFunction
            // When going back from a trampoline we cannot just call cont.resume (stack usage!)
            // We delegate the cont.resumeWith(it) call to runCallLoop
            this.cont = cont
            this.result = it
            @Suppress("UNCHECKED_CAST")
            fun runCallLoop(): R {
                // Note: cont is set to null in DeepRecursiveScopeImpl.resumeWith when the whole computation completes
                val result = this.result
                val cont = this.cont
                return (result as Result<R>).getOrThrow() // done -- final result
                // The order of comparison is important here for that case of rogue class with broken equals
                if (UNDEFINED_RESULT == result) {
                    // call 'function' with 'value' using 'cont' as completion
                    val r = try {
                        // This is block.startCoroutine(this, value, cont)
                        function.startCoroutineUninterceptedOrReturn(this, value, cont)
                    } catch (e: Throwable) {
                        cont.resumeWithException(e)
                    }
                    continue
                }

                // If the function returns without suspension -- calls its continuation immediately
                if (r ==

          Open Source Used In DNAC Platform Arica 10005
COROUTINE_SUSPENDED)
                    cont.resume(r as R)
            } else {
                // we returned from a
                crossFunctionCompletion trampoline -- call resume here
                this.result = UNDEFINED_RESULT // reset
                result back
                    cont.resumeWith(result)
            }
        }
    }
}

/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.internal.InlineOnly

/**
 * Returns a hash code value for the object or zero if the object is `null`.
 * @see Any.hashCode
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun Any?.hashCode(): Int =
    this?.hashCode() ?: 0

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

/**
 * Represents a version of the Kotlin standard library.
 * @see [KotlinVersion] for more info.
 */
@SinceKotlin("1.1")
public class KotlinVersion(val major: Int, val minor: Int, val patch: Int) :
    Comparable<KotlinVersion> {
    override fun equals(other: Any?): Boolean {
        if (this === other) return true
        val otherVersion = (other as? KotlinVersion) ?: return false
        return this.version == otherVersion.version
    }

    override fun compareTo(other: KotlinVersion): Int =
        this.version - other.version

    /**
     * Returns `true` if this
     * version is not less than the version specified
     * with the provided [major] and [minor] components.
     */
    public fun isAtLeast(major: Int, minor: Int): Boolean =
        this.major > major || (this.major == major &&
                this.minor >= minor)

    /**
     * Returns `true` if this
     * version is not less than the version specified
     * with the provided [major], [minor] and [patch] components.
     */
    public fun isAtLeast(major: Int, minor: Int, patch: Int): Boolean =
        this.major > major || (this.major == major &&
                (this.minor > minor || this.minor == minor
                        &&
                    this.patch >= patch))

    companion object {
        /**
         * Maximum value a version
         * component can have, a constant value 255.
         */
        public const val MAX_COMPONENT_VALUE = 255

        /**
         * Returns the current version of the Kotlin standard library.
         */
        @kotlin.jvm.JvmField
        public val CURRENT: KotlinVersion = KotlinVersion(1, 4, 10)
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("LateinitKt")
@file:Suppress("unused")

package kotlin

import kotlin.internal.InlineOnly
import kotlin.internal.AccessibleLateinitPropertyLiteral
import kotlin.reflect.KProperty0

/**
 * Returns `true` if this lateinit property has been assigned a value, and `false`
 * otherwise.
 * Cannot be used in an inline function, to avoid binary compatibility issues.
 */
@SinceKotlin("1.2")
@InlineOnly
inline val @receiver:AccessibleLateinitPropertyLiteral
    KProperty0<*>.isInitialized: Boolean
    get() = throw NotImplementedError("Implementation is intrinsic")

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("LazyKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin

import kotlin.reflect.KProperty

/**
 * Represents a value with lazy initialization.
 * To create an instance of
 * [Lazy] use the [Lazy] function.
 */
public interface Lazy<out T> {
    /**
     * Gets the lazily initialized value of
     * this property.
     */
    val value: T

    companion object {
        val MAX_COMPONENT_VALUE: Int = 255
        val CURRENT: KotlinVersion = KotlinVersion(1, 4, 10)
    }
}

Open Source Used In DNAC Platform Arica 10006
the current Lazy instance.*
   * Once the value was initialized it must not change during the rest of lifetime of this
Lazy instance.*
   */

public fun isInitialized(): Boolean

/**
   * Returns `true` if a value for this Lazy instance has been
   * already initialized, and `false` otherwise.*
   * Once this function has returned `true` it stays `true` for the rest of
lifecycle of this Lazy instance.*
   */

fun lazyOf(value: T): Lazy<T> = InitializedLazyImpl(value)

/**
   * An extension to delegate a read-only property of type [T] to an instance of
[Lazy].
   */

@kotlin.internal.InlineOnly
operator fun <T> Lazy<T>.getValue(thisRef: Any?, property: KProperty<*>): T = value

/**
   * Specifies how a Lazy instance synchronizes initialization
   * among multiple threads.*
   */

public enum class LazyThreadSafetyMode {

    SYNCHRONIZED,

    PUBLICATION,

    NONE,
}

internal object UNINITIALIZED_VALUE

// internal to be called from lazy in JS
internal class UnsafeLazyImpl<out T>(initializer: () -> T) : Lazy<T>, Serializable {
    private var initializer: (() -> T)? = initializer
    private var _value: Any? = UNINITIALIZED_VALUE

    override val value: T
        get() {
            if (_value === UNINITIALIZED_VALUE) {
                _value = initializer!!()
                initializer = null
            }
            return _value
        }

    override fun isInitialized(): Boolean = _value !== UNINITIALIZED_VALUE

    override fun toString(): String = if (isInitialized())
        value.toString() else "Lazy value not initialized yet."

    private fun writeReplace(): Any = InitializedLazyImpl(value)
}

internal class InitializedLazyImpl<out T>(override val value: T) : Lazy<T>, Serializable {

    override fun isInitialized(): Boolean = true

    override fun toString(): String = value.toString()

}
number.rotateLeft(n % 32)\n
Int.rotateLeft(bitCount: Int): Int\n
Int.rotateRight(bitCount: Int): Int

Long.rotateLeft(bitCount: Int): Long

Long.rotateRight(bitCount: Int): Long

Byte.countOneBits(): Int

Byte.countLeadingZeroBits(): Int

Byte.countTrailingZeroBits(): Int

Byte.takeHighestOneBit(): Byte

Byte.takeLowestOneBit(): Byte

/*
 * Counts the number of significant bits on the left side. The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side. Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count. Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally:
 */

number.rotateRight(-n) == number.rotateLeft(n)\n
number.rotateRight(n % 32) == number.rotateLeft(n % 32)

/*
 * Counts the number of most significant bits that are zero in the binary representation of this [Long] number.
 */

Long.countLeadingZeroBits(): Int

/*
 * Counts the number of least significant bits that are zero in the binary representation of this [Long] number.
 */

Long.countTrailingZeroBits(): Int

/*
 * Returns a number having a single bit set in the position of the least significant set bit of this [Long] number, or zero, if this number is zero.
 */

Long.takeHighestOneBit(): Long

Long.takeLowestOneBit(): Long

/*
 * Counts the number of set bits in the binary representation of this [Int] number right by the specified [bitCount] number of bits. The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side. Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count. Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally:
 */

Int.rotateLeft(n % 32) == Int.rotateRight(-n)\n
Int.rotateRight(n % 32) == Int.rotateLeft(-n)

/*
 * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number, or zero, if this number is zero.
 */

Int.rotateLeft(n % 32) == Int.rotateRight(-n)\n
Int.rotateRight(n % 32) == Int.rotateLeft(-n)

/*
 * Counts the number of set bits in the binary representation of this [Byte] number.
 */

Byte.countOneBits(): Int

/*
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Byte] number.
 */

Byte.countLeadingZeroBits(): Int

/*
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Byte] number.
 */

Byte.countTrailingZeroBits(): Int

/*
 * Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number, or zero, if this number is zero.
 */

Byte.takeHighestOneBit(): Byte

/*
 * Returns a number having a single bit set in the position of the least significant set bit of this [Byte] number, or zero, if this number is zero.
 */

Byte.takeLowestOneBit(): Byte
number having a single bit set in the position of the least significant set bit of this [Byte] number, \n * or zero, if this number is zero.\n
@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@kotlin.internal.InlineOnly\npublic inline fun Byte.takeLowestOneBit(): Byte = toInt().takeLowestOneBit().toByte()\n
/**\n * Rotates the binary representation of this [Byte] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * number.rotateLeft(-n) == number.rotateRight(n)\n * Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally:\n * number.rotateLeft(n) == number.rotateLeft(n % 8)\n
@SinceKotlin("1.3")@ExperimentalStdlibApi\npublic fun Byte.rotateRight(bitCount: Int): Byte =\n    (toInt().shl(bitCount and 7) or (toInt() and 0xFF).ushr(8 - (bitCount and 7))).toByte()\n
/**\n * Rotates the binary representation of this [Byte] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * number.rotateRight(-n) == number.rotateLeft(n)\n
@SinceKotlin("1.3")@ExperimentalStdlibApi\npublic fun Byte.rotateLeft(bitCount: Int): Byte =\n    (toInt().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()\n
/**\n * Rotates the binary representation of this [Short] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n * number.rotateLeft(-n) == number.rotateRight(n)\n * Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally:\n * number.rotateLeft(n) == number.rotateLeft(n % 16)\n
@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@kotlin.internal.InlineOnly\npublic inline fun Short.countOneBits(): Int = (toInt() and 0xFFFF).countOneBits()\n
/**\n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Short] number.\n
@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@kotlin.internal.InlineOnly\npublic inline fun Short.countLeadingZeroBits(): Int =\n    (toInt() and 0xFFFF).countLeadingZeroBits() - (Int.SIZE_BITS - Short.SIZE_BITS)\n
/**\n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Short] number.\n
@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@kotlin.internal.InlineOnly\npublic inline fun Short.countTrailingZeroBits(): Int =\n    (toInt() or 0x10000).countTrailingZeroBits()\n
/**\n * Returns a number having a single bit set in the position of the most significant set bit of this [Short] number, \n * or zero, if this number is zero.\n
@SinceKotlin("1.4")@WasExperimental(ExperimentalStdlibApi::class)@kotlin.internal.InlineOnly\npublic inline fun Short.takeHighestOneBit(): Short =\n    toInt().takeHighestOneBit().toShort()\n
/**\n * Rotates the binary representation of this [Short] number right by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the right side reenter the number as the least significant bits on the right side.\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n * number.rotateRight(-n) == number.rotateLeft(n)\n * Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally:\n * number.rotateRight(n % 16)\n
@SinceKotlin("1.3")@ExperimentalStdlibApi\npublic fun Short.rotateRight(bitCount: Int): Short =\n    (toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public fun Short.rotateRight(bitCount: Int): Short =
    (toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()

/*
 * Copyright
2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind

@kotlin.internal.InlineOnly
@SinceKotlin("1.2")
@Suppress("INVISIBLE_MEMBER", "INVISIBLE_REFERENCE")
@RequireKotlin("1.2.30", level = DeprecationLevel.HIDDEN, versionKind = RequireKotlinVersionKind.COMpiler_VERSION)
@public inline fun <R> suspend(noinline block: suspend () -> R): suspend () -> R = block

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("TuplesKt")

package kotlin

/**
 * Represents a generic pair of two values.
 * There is no meaning attached to values in this class, it can be used for any purpose.
 * Pair exhibits value semantics, i.e. two pairs are equal if both components are equal.
 * Example of decomposing it into values:
 * @sample samples.misc.Tuples.pairDestructuring
 *
 * @param A type of the first value.
 * @param B type of the second value.
 * @property first First value.
 * @property second Second value.
 * @constructor
 * Creates a new instance of Pair.
 */
public data class Pair<out A, out B>(
    public val first: A,
    public val second: B
) : Serializable {

    /**
     * Returns string representation of the [Pair] including its [first] and [second] values.
     */
    public override fun toString(): String = "($first, $second)"
}

/**
 * Creates a tuple of type [Pair] from this and [that].
 * This can be useful for creating [Map] literals with less noise, for example:
 * @sample samples.collections.Maps.Instantiation.mapFromPairs
 */
@public infix fun <A, B> A.to(that: B): Pair<A, B> = Pair(this, that)

/**
 * Converts this pair into a list.
 * @sample samples.misc.Tuples.pairToList
 */
@public fun <T> Pair<T, T>.toList(): List<T> = listOf(first, second)

/**
 * Represents a triad of values
 * There is no meaning attached to values in this class, it can be used for any purpose.
 * Triple exhibits value semantics, i.e. two triples are equal if all three components are equal.
 * Example of decomposing it into values:
 * @sample samples.misc.Tuples.tripleDestructuring
 *
 * @param A type of the first value.
 * @param B type of the second value.
 * @param C type of the third value.
 */
public data class Triple<out A, out B, out C>(
    public val first: A,
    public val second: B,
    public val third: C
) : Serializable {

    /**
     * Returns string representation of the [Triple] including its [first], [second] and [third] values.
     */
    public override fun toString(): String = "($first, $second, $third)"
}

/**
 * Converts this triple into a list.
 * @sample samples.misc.Tuples.tripleToList
 */
@public fun <T> Triple<T, T, T>.toList(): List<T> = listOf(first, second, third)

/*
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin.ranges

import kotlin.internal.

/**
 * A range of values of type `UInt`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public class UIntRange(start: UInt, endInclusive: UInt)
    : UIntProgression(start, endInclusive, 1), ClosedRange<UInt> {
    override val start: UInt get() = first
    override val endInclusive: UInt get() = last

    override fun contains(value: UInt): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean =
        first == endInclusive

    override fun equals(other: Any?): Boolean =
        other is UIntRange && (isEmpty() && other.isEmpty() ||
            first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())

    override fun toString(): String = "$first..$last"

    companion object {
        /** An empty range of values of type UInt. */
        @public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)
    }
}

/**
 * A progression of values of type `UInt`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public class UIntProgression
internal constructor(
    start: UInt,
    endInclusive: UInt,
    step: Int
) : Iterable<UInt> { /*...*/
    init {
        if (step == 0) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE

Open Source Used In DNAC Platform Arica 10010
to avoid overflow on negation.\n
The first element in the progression.\n
The last element in the progression.\n
The step of the progression.\n
Checks if the progression is empty.\n
public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n
override fun equals(other: Any?): Boolean = other is UIntProgression && isEmpty()\n
override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step.toInt())\n
companion object {\n
/**
 * Creates UIntProgression within the specified bounds of a closed range.\n * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].\n * In order to go backwards the [step] must be negative.\n * In the case step must be greater than Int.MIN_VALUE and not equal to zero.\n
* An iterator over a progression of values of type `UInt`.\n * @property step the number by which the value is incremented on each step.\n * @property start the first element in the progression.\n * @property last the last element in the progression.\n * @property step the step of the progression.\n
/**\n * The first element in the progression.\n */
public val first: UInt = start\n
/**\n * The last element in the progression.\n */
public val last: UInt = getProgressionLastElement(start, endInclusive, step)\n
/**\n * The step of the progression.\n */
public var step: Int = step

override fun iterator(): UIntIterator = UIntProgressionIterator(first, last, step)\n
/**\n * Checks if the progression is empty.\n */
public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last\n
override fun equals(other: Any?): Boolean = other is UIntProgression && isEmpty()\n
override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step.toInt())
 ULonRange(ULong.MAX_VALUE, ULong.MIN_VALUE)\n  |\n  |**\n  * A progression of values of type 'ULong'.\n*\n  * @SinceKotlin('1.3')\n*\n  * An internal constructor.\n  *\n  * @start: ULong.\nendInclusive: ULong.\nstep: Long\n  :\n  \n    Iterable<ULong> \n    |\n    | when (step == 0) throw kotlin.IllegalArgumentException("Step must be non-zero.");\n    |\n    | when (step == ULong.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than ULong.MIN_VALUE to avoid overflow on negation.");\n  |\n  |**\n  * The first element in the progression.\n  *\n  * @public val first: ULong = start\n  |\n  |**\n  * The last element in the progression.\n  *\n  * @public val last: ULong = getProgressionLastElement(start, endInclusive, step)\n  |\n  |**\n  * The step of the progression.\n  *\n  * @public val step: Long = step\n  |\n  | override fun iterator(): ULongIterator =\n  |\n  | ULongProgressionIterator(first, last, step)\n  |**\n  * Checks if the progression is empty.\n  *\n  * @@isEmpty(): Boolean = if (step > 0) first > last else false\n  |\n  | override fun equals(other: Any?): Boolean =\n  |\n  | other is ULongProgression && (isEmpty() && other.isEmpty() ||\n  | (first == other.first && last == other.last && step == other.step)\n  |**\n  * Override fun hashCode().\n  *\n  * @@toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step $(-step)"\n  |\n  |**\n  * Creates ULongProgression within the specified bounds of a closed range.\n  |\n  * @fromClosedRange(rangeStart: ULong, rangeEnd: ULong, step: Long): ULongProgression = ULongProgression(rangeStart, rangeEnd, step)\n  |\n  |**\n  * An iterator over a progression of values of type 'ULong'.\n  *\n  * @step the number by which the value is incremented on each step.\n  *\n  * @property step: Long = step\n  |\n  |**\n  * In order to go backwards the [step] must be negative.\n  *\n  * @hasNext(): Boolean = if (step > 0) first <= last\n  |\n  | override fun fun hashCode(): Int =\n  |\n  | (isEmpty() && return $(-1) * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())\n  |\n  |**\n  * Returns the smaller of two values.\n  *\n  * @min(a: UInt, b: UInt): UInt = minOf(a, b)\n  |\n  |**\n  * Returns the greater of two values.\n  *\n  * @max(a: UInt, b: UInt): UInt = maxOf(a, b)\n  |\n  |**\n  * Counts the number of set bits in the binary representation of this [UInt].\n  *\n  * @countOneBits(): Int = toInt().countOneBits()\n  |\n  |**\n  * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt].\n  *\n  * @countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()\n  |\n  |**\n  * Counts the number of consecutive least significant bits that are zero in the
binary representation of this [UInt] number.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countTrailingZeroBits(): Int =
toInt().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of
 * the most significant set bit of this [UInt] number, n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeHighestOneBit(): UInt =
toInt().takeHighestOneBit().toUInt()

/**
 * Returns a number having a single bit set in the position of
 * the least significant set bit of this [UInt] number, n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@WasExperimental(ExperimentalStdlibApi::class)
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeLowestOneBit(): UInt =
toInt().takeLowestOneBit().toUInt()

/**
 * Rotates the binary representation of this [UInt] number left by
 * the specified [bitCount] number of bits. n * The most significant bits pushed out from the left side reenter the number as
 * the least significant bits on the right side. n * Rotating the number left by a negative bit count is the same as
 * rotating it right by the negated bit count: n * number.rotateLeft(-n) == number.rotateRight(n)
 * Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally
 * number.rotateLeft(n % 32)
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateLeft(bitCount: Int): UInt =
toInt().rotateLeft(bitCount).toUInt()

/**
 * Rotates the binary representation of this [UInt] number right by
 * the specified [bitCount] number of bits. n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count: n *
 * number.rotateRight(-n) == number.rotateLeft(n)
 * Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally
 * number.rotateRight(n % 32)
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateRight(bitCount: Int): UInt =
toInt().rotateRight(bitCount).toUInt()

/**
 * Counts the number of set bits in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countOneBits(): Int =
toLong().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in
 * the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countLeadingZeroBits(): Int =
toLong().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in
 * the binary representation of this [ULong] number.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.countTrailingZeroBits(): Int =
toLong().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of
 * the most significant set bit of this [ULong] number, n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeHighestOneBit(): ULong =
toInt().takeHighestOneBit().toULong()

/**
 * Returns a number having a single bit set in the position of
 * the least significant set bit of this [ULong] number, n * or zero, if this number is zero.
 */
@SinceKotlin("1.4")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULong.takeLowestOneBit(): ULong =
toInt().takeLowestOneBit().toULong()
same as rotating it right by the negated bit count:
* number.rotateLeft(-n) == number.rotateRight(n)
* Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateLeft(n) == number.rotateLeft(n % 64)
* Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits.
* The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
* number.rotateRight(-n) == number.rotateLeft(n) *n * Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:
* number.rotateRight(n) == number.rotateRight(n % 64)
* Rotates the number right by a negative bit count is the same as rotating it left by the negated bit count:
@kotlin.internal.InlineOnly
public inline fun UShort.countOneBits(): Int = toUInt().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ USHORT ] number.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ USHORT ] number.
 */
@SinceKotlin("1.4")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()

/**
 * Rotates the binary representation of this [ USHORT ] number left by the specified [ bitCount ] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()

/**
 * Rotates the binary representation of this [ USHORT ] number right by the specified [ bitCount ] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()

// (a - b) mod c
private fun differenceModulo(a: UInt, b: UInt, c: UInt): UInt {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

private fun differenceModulo(a: ULong, b: ULong, c: ULong): ULong {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range [ start ] to [ end ] in case of a positive [ step ], or from [ end ] to [ start ] in case of a negative [ step ].
 * No validation on passed parameters is performed. The given parameters should satisfy the condition:
 * - either ` step > 0 ` and ` start <= end `, ` start <= end ` or ` step < 0 ` and ` start >= end `.
 * - @param start first element of the progression
 * - @param end ending bound for the progression
 * - @param step increment, or difference of successive elements in the progression
 */
@PublishedApi
@SinceKotlin("1.3")
internal fun getProgressionLastElement(start: UInt, end: UInt, step: Int): UInt = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toUInt())
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toUInt())
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}
arithmetic progression, i.e. the last element of the progression which is in the range \(n\) from \([start]\) to \([end]\) in case of a positive \([step]\), or from \([end]\) to \([start]\) in case of a negative \([step]\). \(n\) * \(n\) * No validation on passed parameters is performed. The given parameters should satisfy the condition: \(n\) * \(n\) * - either \(step > 0\) and \(start \leq end\), \(n\) * - or \(step < 0\) and \(start \geq end\), \(n\) * \(n\) * @param start first element of the progression \(n\) * @param end

ending bound for the progression \(n\) * @param step increment, or difference of successive elements in the progression \(n\) * @return the final element of the progression \(n\) * @suppress

*/
@PublishedApi
@SinceKotlin("1.3")
internal fun getProgressionLastElement(start: ULong, end: ULong, step: Long): ULong = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toULong())
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toULong())
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

@file:kotlin.jvm.JvmName("UStringsKt") // string representation of unsigned numbers

package kotlin.text

/**
 * Returns a string representation of this \[Byte\] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public /*inline*/ fun UByte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this \[Short\] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public /*inline*/ fun UShort.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this \[Int\] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
//@kotlin.internal.InlineOnly
public /*inline*/ fun UInt.toString(radix: Int): String = this.toLong().toString(radix)

/**
 * Returns a string representation of this \[Long\] value in the specified [radix].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.toString(radix: Int): String = ulongToString(this.toLong(), checkRadix(radix))

/**
 * Parses the string as a signed \[UByte\] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a signed \[UByte\] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(radix: Int): UByte = toUByteOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an \[UInt\] number and returns the result.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUInt(): UInt = toUIntOrNull() ?: numberFormatError(this)

/**
 * Parses the string as an \[UInt\] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUInt(radix: Int): UInt =
toUIntOrNull(radix) ?: numberFormatError(this)/\n
/**
 * Parses the string as a [ULong] number and returns the
 * result
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULong(): ULong =
toULongOrNull() ?: numberFormatError(this)/\n
/**
 * Parses the string as a [ULong] number and returns the result
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(): ULong? =
toULongOrNull(radix) ?: numberFormatError(this)/\n
/**
 * Parses the string as a [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(radix: Int): ULong? =
toULongOrNull(radix) ?: numberFormatError(this)/\n
/**
 * Parses the string as a [UByte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByteOrNull(): UByte? =
toUByteOrNull(radix = 10) ?: numberFormatError(this)/\n
/**
 * Parses the string as a [UByte] number and returns the result.
 * @throws NumberFormatException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByteOrNull(radix: Int): UByte? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int > UByte.MAX_VALUE) return null
    return int.toUByte()
}

/**
 * Parses the string as a [UShort] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShortOrNull(): UShort? =
toUShortOrNull(radix = 10) ?: numberFormatError(this)/\n
/**
 * Parses the string as a [UShort] number and returns the result
 * @throws NumberFormatException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShortOrNull(radix: Int): UShort? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val limit: UShort = UShort.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 512409557603043100uL  //  limit / 36

    var limitBeforeMul = limitForMaxRadix
    val uradix = radix.toULong()
    var result = 0uL
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {

                limitBeforeMul = limitForMaxRadix / uradix
                val uradix =
toULong()
                var result = 0uL
                for (i in start until length) {
                    val digit = digitOf(this[i], radix)
                    if (digit < 0) return null
                    if (result > limitBeforeMul) {
                        return null
}
limitBeforeMul = limit / uradix

if (result > limitBeforeMul) {
    return null
}

return result

result *= uradix

val beforeAdding = result

result += digit.toUInt()

if (result < beforeAdding) return null // overflow has happened

return result

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:Suppress("INVISIBLE_REFERENCE", "INVISIBLE_MEMBER")
package kotlin

import kotlin.annotation.AnnotationTarget.*
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind

/**
 * Marks the API that is dependent on the experimental unsigned types, including those types themselves.
 * Usages of such API will be reported as warnings unless an explicit opt-in within * the [OptIn] annotation, e.g. `@OptIn(ExperimentalUnsignedTypes::class)`, * or with the `-Xopt-in=kotlin.ExperimentalUnsignedTypes` compiler option is given.
 * It's recommended to propagate the experimental status to the API that depends on unsigned types by annotating it with this annotation.
 */
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MathKt")

package kotlin.math

/** Ratio of the circumference of a circle to its diameter, approximately 3.14159. */
@SinceKotlin("1.2")
public const val PI: Double = 3.141592653589793
/** Base of the natural logarithms, approximately 2.71828. */
@SinceKotlin("1.2")
public const val E: Double = 2.718281828459045

region ==-

// region ================
Double Math ========================================

/** Computes the sine of the angle [x] given in radians. */
@SinceKotlin("1.2")
public expect fun sin(x: Double): Double

/** Computes the cosine of the angle [x] given in radians. */
@SinceKotlin("1.2")
public expect fun cos(x: Double): Double

/** Computes the tangent of the angle [x] given in radians. */
@SinceKotlin("1.2")
public expect fun tan(x: Double): Double

/** Computes the arc sine of the value [x]; the returned value is an angle in the range from `-PI/2` to `+PI/2` radians. */
@SinceKotlin("1.2")
public expect fun asin(x: Double): Double

/** Computes the arc cosine of the value [x]; the returned value is an angle in the range from `0.0` to `+PI` radians. */
@SinceKotlin("1.2")
public expect fun acos(x: Double): Double

/** Computes the arc tangent of the value [x]; the returned value is an angle in the range from `-PI/2` to `+PI/2` radians. */
@SinceKotlin("1.2")
public expect fun atan(x: Double): Double

/** Computes the hyperbolic sine of the value [x]. */
@SinceKotlin("1.2")
public expect fun sinh(x: Double): Double

/** Computes the hyperbolic cosine of the value [x]. */
@SinceKotlin("1.2")
public expect fun cosh(x: Double): Double

/** Computes the hyperbolic tangent of the value [x]. */
@SinceKotlin("1.2")
public expect fun tanh(x: Double): Double

/** Computes the hyperbolic arc sine of the value [x]; the returned value is an angle in the range from `-PI/2` to `+PI/2` radians. */
@SinceKotlin("1.2")
public expect fun asinh(x: Double): Double

/** Computes the hyperbolic arc cosine of the value [x]; the returned value is an angle in the range from `0.0` to `+PI` radians. */
@SinceKotlin("1.2")
public expect fun acosh(x: Double): Double

/** Computes the hyperbolic arc tangent of the value [x]; the returned value is an angle in the range from `-PI/2` to `+PI/2` radians. */
@SinceKotlin("1.2")
public expect fun atanh(x: Double): Double

Open Source Used In DNAC Platform Arica 10018
- `sinh(-Inf)` is `-Inf`

```kotlin
@SinceKotlin("1.2")
public expect fun sinh(x: Double): Double
```

- `cosh(NaN)` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun cosh(x: Double): Double
```

- `tanh(NaN)` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun tanh(x: Double): Double
```

- `asinh(NaN)` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun asinh(x: Double): Double
```

- `acosh(NaN)` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun acosh(x: Double): Double
```

- `atanh(NaN)` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun atanh(x: Double): Double
```

- `sqrt(x^2 + y^2)` without intermediate overflow or underflow.

```kotlin
@SinceKotlin("1.2")
public expect fun hypot(x: Double, y: Double): Double
```

- `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun sqrt(x: Double): Double
```

- `exp(x)` is `NaN` when `x < 0` or `x` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun exp(x: Double): Double
```

- `expm1(x)` is `NaN` when `x < -1.0`

```kotlin
@SinceKotlin("1.2")
public expect fun expm1(x: Double): Double
```

- `ln(x)` is `NaN` when `x < 0` or `x` is `NaN`

```kotlin
@SinceKotlin("1.2")
public expect fun ln(x: Double): Double
```

- `log(x, b)` is `NaN` when `x < 0` or `b` is not a positive integer.

```kotlin
@SinceKotlin("1.2")
public expect fun log(x: Double, base: Double): Double
```

- `1.0` is `NaN` when `x < 1`

```kotlin
@SinceKotlin("1.2")
public expect fun Floor(x: Double): Double
```

- `0.0` is `NaN` when `x < 0`

```kotlin
@SinceKotlin("1.2")
public expect fun Ceil(x: Double): Double
```

- `-1.0` is `NaN` when `x > 1`

```kotlin
@SinceKotlin("1.2")
public expect fun Floor(x: Double): Double
```
Double\n/** 
 * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.\n * @see absoluteValue
 */
public expect fun round(x: Double): Double

/** 
 * Returns the sign of the given value [x]:
 *   - `-1.0` if the value is negative,\n *   - zero if the value is zero,\n *   - `1.0` if the value is positive\n */
public expect fun sign(x: Double): Double

/** 
 * Returns the smaller of two values.\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun min(a: Double, b: Double): Double

/** 
 * Returns the greater of two values.\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun max(a: Double, b: Double): Double

/** 
 * Returns the absolute value of this value.\n * @see abs function\n */
public expect val absoluteValue: Double

/** 
 * Returns the ulp (unit in the last place) of this value.\n * @see ulp function\n */
public expect val ulp: Double

/** 
 * Returns the value nearest to this value in direction of positive infinity.\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun nextUp(): Double

/** 
 * Returns the value nearest to this value in direction of negative infinity.\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun nextDown(): Double

/** 
 * Returns the value nearest to this value in direction from this value towards the value [to].\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun nextTowards(to: Double): Double

/** 
 * Rounds the [Double] value nearest to the nearest integer and converts the result to [Int].\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun roundToInt(): Int

/** 
 * Rounds this [Double] value nearest to the nearest integer and converts the result to [Long].\n * @throws IllegalArgumentException when this value is `NaN`\n */
public expect fun roundToLong(): Long

// endregion

// region ================ Float Math
========================================

/** Computes the sine of the angle [x] given in radians.\n * @see absoluteValue\n */
public expect fun sin(x: Float): Float

/** Computes the cosine of the angle [x] given in radians.\n * @see absoluteValue\n */
public expect fun cos(x: Float): Float

/** Computes the tangent of the angle [x] given in radians.\n * @see absoluteValue\n */
public expect fun tan(x: Float): Float

// endregion
*///
@SinceKotlin("1.2")
public expect fun tan(x: Float): Float

/**
 * Computes the arc cosine of the value [x];
 * the returned value is an angle in the range from `0.0`
 * to `PI` radians.
 * Special cases:
 * - `acos(x)` is `NaN` when `abs(x)` > 1 or x is `NaN`
 */
@SinceKotlin("1.2")
public expect fun acos(x: Float): Float

/**
 * Returns the angle 'theta' of the polar coordinates (r, theta) that correspond
 * to the rectangular coordinates (x, y) by computing the arc tangent of the
 * value [y] / [x]; the returned value is an angle in the range from `-PI` to `PI`
 * radians.
 * Special cases:
 * - `atan(NaN)` is `NaN`
 * - `atan2(0.0, 0.0)` is `0.0`
 * - `atan2(0.0, x)` is `0.0` for `x > 0` and `PI` for `x < 0`
 * - `atan2(y, +Inf)` is `0.0` for `y < 0` and `-PI` for `y > 0`
 * - `atan2(y, -Inf)` is `-PI` for `y < 0` and `PI` for `y > 0`
 * - `atan2(-Inf, x)` is `-PI` for `x < 0` and `PI` for `x > 0`
 * - `atan2(+Inf, x)` is `PI` for `x < 0` and `-PI` for `x > 0`
 */
@SinceKotlin("1.2")
public expect fun atan2(y: Float, x: Float): Float

/**
 * Computes the hyperbolic sine of the value [x].
 * Special cases:
 * - `sinh(NaN)` is `NaN`
 * - `sinh(+Inf)` is `+Inf`
 * - `sinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun sinh(x: Float): Float

/**
 * Computes the hyperbolic cosine of the value [x].
 * Special cases:
 * - `cosh(NaN)` is `NaN`
 * - `cosh(+Inf)` is `+Inf`
 */
@SinceKotlin("1.2")
public expect fun cosh(x: Float): Float

/**
 * Computes the hyperbolic tangent of the value [x].
 * Special cases:
 * - `tanh(NaN)` is `NaN`
 * - `tanh(+Inf)` is `1.0`
 * - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
public expect fun tanh(x: Float): Float

/**
 * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.
 * Special cases:
 * - returns `+Inf` if any of arguments is infinite
 * - returns `NaN` if any of arguments is `NaN` and the other is not infinite
 */
@SinceKotlin("1.2")
public expect fun hypot(x: Float, y: Float): Float

/**
 * Computes Euler's number `e` raised to the power of the value [x].
 * Special cases:
 * - `exp(NaN)` is `NaN`
 * - `exp(+Inf)` is `+Inf`
 * - `exp(-Inf)` is `0.0`
 */
@SinceKotlin("1.2")
public expect fun exp(x: Float): Float

/**
 * Computes `exp(x) - 1.0`.
 */
@SinceKotlin("1.2")
public expect fun expm1(x: Float): Float

/**
 * Computes the natural logarithm (base `E`) of the value [x].
 * Special cases:
 * - `ln(NaN)` is `NaN`
 * - `ln(x)` is `NaN` when `x < 0`
 */
@SinceKotlin("1.2")
public expect fun ln(x: Float): Float
Float): Float

/**
 * Computes the common logarithm (base 10) of the value \[x\].
 */
@SinceKotlin("1.2")
public expect fun log10(x: Float): Float

/**
 * Computes the binary logarithm (base 2) of the value \[x\].
 */
@SinceKotlin("1.2")
public expect fun log2(x: Float): Float

/**
 * Computes \(\ln(x + 1)\). This function can be implemented to produce more precise result for \[x\] near zero.
 */
@SinceKotlin("1.2")
public expect fun ln1p(x: Float): Float

/**
 * Rounds the given value \[x\] to an integer towards positive infinity.
 */
@SinceKotlin("1.2")
public expect fun ceil(x: Float): Float

/**
 * Rounds the given value \[x\] to an integer towards negative infinity.
 */
@SinceKotlin("1.2")
public expect fun floor(x: Float): Float

/**
 * Rounds the given value \[x\] to an integer towards zero.
 */
@SinceKotlin("1.2")
public expect fun truncate(x: Float): Float

/**
 * Rounds the given value \[x\] towards the closest integer with ties rounded towards even integer.
 */
@SinceKotlin("1.2")
public expect fun round(x: Float): Float

/**
 * Returns the absolute value of the given value \[x\].
 */
@SinceKotlin("1.2")
public expect fun abs(x: Float): Float

/**
 * Returns the sign of the given value \[x\]:
 *   - `-1.0` if the value is negative,
 *   - zero if the value is zero,
 *   - `1.0` if the value is positive
 */
@SinceKotlin("1.2")
public expect fun sign(x: Float): Float

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public expect fun min(a: Float, b: Float): Float

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public expect fun max(a: Float, b: Float): Float

// extensions

/**
 * Raises this value to the power \[x\].
 */
@SinceKotlin("1.2")
public expect fun Float.pow(x: Float): Float

/**
 * Raises this value to the integer power \[n\].
 */
@SinceKotlin("1.2")
public expect fun Float.pow(n: Int): Float

/**
 * Returns this value with the sign bit same as of the \[sign\] value.
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Float): Float

/**
 * Returns this value with the sign bit same as of the \[sign\] value.
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Int): Float

/**
 * Rounds this \[Float\] value to the nearest integer and converts the result to \[Int\].
 */
@SinceKotlin("1.2")
public expect fun Float.roundToInt(): Int

/**
 * Rounds this \[Float\] value to the nearest integer and converts the result to \[Long\].
 */
@SinceKotlin("1.2")
public expect fun Float.roundToLong(): Long

Open Source Used In DNAC Platform Arica 10022
//*@SinceKotlin("1.2")*public expect fun Float.roundToLong(): Long

============= Integer Math ==============================================================

the absolute value of the given value \(n\), \(n\ *\) * Special cases:
\(n\ *\ -`abs(Int.MIN_VALUE)` is 'Int.MIN_VALUE' due to an overflow
\(n\ *\ @see absoluteValue extension property for [Int]\)

\(*@SinceKotlin("1.2")*public expect fun abs(n: Int): Int

\/**
 * Returns the absolute value of the given value \(n\), \(n\ *\ - Special cases:
 * - `abs(Int.MIN_VALUE)` is 'Int.MIN_VALUE' due to an overflow
 * @see abs function
 */
@SinceKotlin("1.2")
public expect val Int.absoluteValue: Int

\/**
 * Returns the sign of this value:
 * - `-1` if the value is negative, \(n\ *\ -0` if the value is zero, \(n\ *\ -1` if the value is positive
 * @see abs function
 */
@SinceKotlin("1.2")
public expect val Int.sign: Int

// endregion

// region

// endregion

// region


/**
 * Returns the absolute value of the given value \(n\), \(n\ *\ - Special cases:
 * - `abs(Long.MIN_VALUE)` is 'Long.MIN_VALUE' due to an overflow
 * @see absoluteValue extension property for [Long]\)

\(*@SinceKotlin("1.2")*public expect fun abs(n: Long): Long

\/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public expect fun min(a: Long, b: Long): Long

\/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public expect fun max(a: Long, b: Long): Long

\/**
 * Returns the absolute value of this value.
 *
 * Special cases:
 * - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow
 * @see abs function
 */
@SinceKotlin("1.2")
public expect val Long.absoluteValue: Long

\/**
 * Returns the sign of this value:
 * - `-1` if the value is negative, \(n\ *\ -0` if the value is zero, \(n\ *\ -1` if the value is positive
 * @see abs function
 */
@SinceKotlin("1.2")
public expect val Long.sign: Int

// endregion

Open Source Used In DNAC Platform Arica

10035
Open Source Used In DNAC Platform Arica

10049
Open Source Used In DNAC Platform Arica 10050
Open Source Used In DNAC Platform Articles

10053
Open Source Used In DNAC Platform Arica 10089
Open Source Used In DNAC Platform Arica 10092
Open Source Used In DNAC Platform

Arca 10097
Open Source Used In DNAC Platform Arica 10100
Open Source Used In DNAC Platform Arica 10105

kB, bAAmB, cAAnB, EAAmB, sBAAnB, UAhjCIB, EAgjC+C, IAhjC+C, CAAlJ, C; YAA2C, sBAgjCQ, IAhjC+C++QA
E/C, OAAO, W; O; KAXX, c; wGAcA, yB; MAAa, +D; MAAa, sC; QAMW, kBAmB, gB; QASV, Q; QAAA, 2B; QAaB
H, QAaB, bAAmB, cAAnB, UAhjCIB, EAgjC+C, IAhjC+C, CAAlJ, C; YAA2C, sBAgjCQ, IAhjC+C++QA
E/C, OAAO, W; K; 8FAGX, 6C; MAAmB, QAaB, MAAa, sC; QAMW, kBAmB, gB; QASV, Q; QAAA, 2B; QAaB, WA
AY, WAAI, OAAI, C; MACpD, OAAO, W; K; 0FAGX, 6C; MAAmB, QAaB, MAAa, sC; QAMW, kBAmB, gB; QASV,
Open Source Used In DNAC Platform Arica 10117
Open Source Used In DNAC Platform Arica 10154
Open Source Used in DNAC Platform Arica 10160
Open Source Used In DNAC Platform Arica 10195
Q;MAmRP,OAnRO,2BAQyC,mBAAY,cA2Q3B,KA3Q2B,IAAZ,CARzC,4D;K;uCAsRX,0B;MACI,oCAAa,2BA
AkB,KAAlB,EAAyB,SAAzB,C;MAEb,oBAjRgD,mBAAY,cAiR1B,KAjR0B,IAAZ,C;MARzC,Q;MA0RP,iBA1R
Q;MA0RsB,aA1RtB,4D;MA2RP,mBAAY,aAAZ,IAA6B,O;MAE7B,OAAO,U;K;0CAGX,mB;MAAd0,0BAAQ,
OAA,MAoB,E;K;CAExE,mB;MAisB,IAIA,IAJAI,EAuB,M;MAPzC,Wa3RGd,mBAAY,cA2RnC,S3RmC,IA
AZ,MA6RhD,IAAI,cAAO,IAAX,C;QACI,iBAc,WAAd,UAyB,IAAzB,U;UACLIAIA1,gBAW,mBAAY,KAA
AZ,CAAX,CAAI,C;YAAMc,OAAO,QAQQ,WAARR,1;:aAE3C,IAALeAAQ,IAAZ,C;QACW,kB;QAAuB,sAA
k,BAAY,O;QAArC,qD;UACLIAI1,gBAW,mBAAY,OAAB,mCAAY,CAAX,CAAI,C;YAAMc,OAAO,UAQQ,WAARR,1;
QAE9C,mBAAC,CAAI,UAAS,B,IAIA,UAAC,B;YACI1Al1,gBAW,mBAAY,OAAB,mCAAY,CAAX,CAAI,C;YAAMc,
OAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,MAAC,B;YACI1Al1,gBAW,mBAAY,OAAB,mCAAY,CAAX,
CAAI,C;YAAMc,OAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,
QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,
OAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,IAAI1,cAAO,QAQQ,WAARR,1;:aAE3C,
Open Source Used In DNAC Platform Arica 10210
Open Source Used In DNAC Platform Arica 10242
1.2057 kerby-xdr 1.0.0

1.2057.1 Available under license:

Kerby XDR Project
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2058 caller-path 0.1.0

1.2058.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.2059 babel-helper-define-map 6.26.0 6.26.0
1.2059.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2060 eslint-config-airbnb-base 9.0.0
1.2060.1 Available under license:
MIT

1.2061 fill-range 7.0.1 7.0.1
1.2061.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2062 mime 1.4.1 1.4.1
1.2062.1 Available under license:
The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2063 minimist 1.2.0 1.2.0
1.2063.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2064 is-windows 1.0.2 1.0.2

1.2064.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2065 @babel/helpers 7.16.3

1.2065.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2066 @babel/template 7.16.0
1.2066.1 Available under license :
   MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2067 pause-stream 0.0.11 0.0.11
1.2067.1 Available under license:

Dual Licensed MIT and Apache 2

The MIT License

Copyright (c) 2013 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2013 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2068 for-own 0.1.5
1.2068.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2069 bouncy-castle 1.60
1.2069.1 Available under license:

1) c) 2000-2017 The Legion of the Bouncy Castle Inc.
(http://www.bouncycastle.org) DEALINGS IN THE SOFTWARE.
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2070 lodash.curry 4.1.1 4.1.1

1.2070.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2071 xmlgraphics-commons 2.4

1.2071.1 Available under license:
Apache XML Graphics Commons
Copyright 2006-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2072 jetty-continuation 9.4.41.v20210516
1.2072.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

“Contributor” means any person or entity that distributes the Program.

“Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly
perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor’s responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to
time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

===================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
===================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd 
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

  * The Eclipse Public 1.0 License
Jetty may be distributed under either license.

-----

**Eclipse**

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

**Oracle**

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL/GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

**Oracle OpenJDK**

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

**OW2**

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html
org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by
Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.2073 har-validator 5.0.3 5.0.3
1.2073.1 Available under license:
Copyright (c) 2015, Ahmad Nassri <ahmad@ahmadnassri.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2074 json-schema-compatibility 1.1.0 1.1.0
1.2074.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2075 is-glob 4.0.3

1.2075.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2076 ansi-colors 3.2.3

1.2076.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-present, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2077 jul-to-slf4j 1.7.5

1.2078 d 1.0.1
1.2078.1 Available under license :
   ISC License

Copyright (c) 2013-2019, Mariusz Nowak, @medikoo, medikoo.com

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2079 quartz 2.3.2
1.2079.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of
fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any
medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2080 jersey-hk2 2.32

1.2080.1 Available under license:

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed
source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
   where such changes and/or additions to the Program originate from
   and are Distributed by that particular Contributor. A Contribution
   "originates" from a Contributor if it was added to the Program by
   such Contributor itself or anyone acting on such Contributor's behalf.
   Contributions do not include changes or additions to the Program that
   are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.
"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual
property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.
3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this
Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original
authors’ reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The “Program”, below, refers to any such program or work, and a “work based on the Program” means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   
   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

If distribution of executable or object code is made by offering access
to copy from a designated place, then offering equivalent access to copy
the source code from the same place counts as distribution of the source
code, even though third parties are not compelled to copy the source
along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will
automatically terminate your rights under this License. However, parties
who have received copies, or rights, from you under this License will
not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all
its terms and conditions for copying, distributing or modifying the
Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions
on the recipients’ exercise of the rights granted herein. You are not
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
    `show w'. This is free software, and you are welcome to redistribute
    it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the
    program `Gnomovision' (which makes passes at compilers) written by
    James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

---

1.2081 batik-swing 1.13

1.2081.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software's
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2082 xmlenc 0.52
1.2082.1 Available under license :
BSD-2-Clause

1.2083 javax-inject 2.5.0-b32
1.2083.1 Available under license :
CDDL + GPLv2 with classpath exception
https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
A business-friendly OSS license

1.2084 drafter.js 2.6.7 2.6.7
1.2084.1 Available under license :
(The MIT License)

Copyright (c) 2015 Apiary Inc. <support@apiary.io>.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2085 is-accessor-descriptor 0.1.6 0.1.6
1.2085.1 Available under license :
The MIT License (MIT)
Copyright (c) 2015, Jon Schlinkert.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2086 backport-util-concurrent 3.1
1.2086.1 Available under license :
The backport-util-concurrent software has been released to the public domain,
as explained at: http://creativecommons.org/licenses/publicdomain.
Acknowledgements: backport-util-concurrent is based in large part on the public
domain sources from:
1) JSR166,
2) package dl.util.concurrent,
3) Doug Lea's "collections" package.
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html><head><title>Creative Commons Public Domain</title>
<body>
<p align="center"><em>Copyright-Only Dedication (based on United States law) or Public Domain
Certification</em></p>
<p>The

Open Source Used In DNAC Platform Arica 10304


person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.</p>

<p>A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.</p>

<p>Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.</p>

<p>Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.</p>

1.2087 commons-http-client 3.1
1.2087.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: HOW TO APPLY THE APACHE LICENSE TO YOUR WORK
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2088 avro-ipc 1.9.1
1.2088.1 Available under license:

Apache Avro IPC
Copyright 2009-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

Open Source Used In DNAC Platform Arica 10312
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2089 cookie 0.3.1 0.3.1

1.2089.1 Available under license : (The MIT License)

Copyright (c) 2012-2014 Roman Shytlman <shtylman@gmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2090 bl 2.2.1 2.2.1

1.2090.1 Available under license :

The MIT License (MIT)

===================================

Copyright (c) 2013-2018 bl contributors

===================================

*bl contributors listed at <https://github.com/rvagg/bl#contributors>*
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2091 which 2.0.2
1.2091.1 Available under license :
MIT License

Copyright (c) 2017 Cdric Belin <cedric@belin.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2092 secure-keys 1.0.0 1.0.0
1.2092.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2093 datanucleus-rdbms 4.1.19
1.2093.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2008-2013 DataNucleus

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License, ==
==  Version 2.0, in this case for the DataNucleus distribution.       ==
=====================================================================

This product includes software developed by many individuals, including the following:

Andy Jefferson
Erik Bengtson
Joerg von Frantzius
Marco Schulze

=====================================================================
This product has included contributions from some individuals, including the following:

Barry Haddow
Ralph Ullrich
David Ezzio
Brendan de Beer
David Eaves
Martin Taal
This product also includes software developed by the TJDO project (http://tjdo.sourceforge.net/).

This product also includes software developed by the Apache Commons project (http://commons.apache.org/).

1.2094 restore-cursor 1.0.1

1.2094.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2095 uuid 3.4.0 3.4.0

1.2095.1 Available under license:

The MIT License (MIT)

Copyright (c) 2010-2016 Robert Kieffer and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2096 regenerator-transform 0.10.1

1.2096.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 *
 * This source code is licensed under the BSD-style license found in the
 * https://raw.github.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */

Found in path(s):
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/meta.js
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 *
 * This source code is licensed under the BSD-style license found in the
 * https://raw.github.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */
Found in path(s):
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/leap.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/hoist.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/index.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/visit.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/emit.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/hoist.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/meta.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/src/util.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/visit.js
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 *
 * This source code is licensed under the BSD-style license found in the
 * https://raw.githubusercontent.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */

Found in path(s):
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/leap.js
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/util.js
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 * 
 * This source code is licensed under the BSD-style license found in the
 * https://raw.githubusercontent.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */

Found in path(s):
* /opt/cola/permits/1110631362_1606842466.21/0/regenerator-transform-0-10-1-5-tgz/package/lib/emit.js

1.2097 babel-core 6.26.3
1.2097.1 Available under license :
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2098 sax 1.2.4 1.2.4

1.2098.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

=====

`String.fromCodePoint` by Mathias Bynens used according to terms of MIT License, as follows:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2099 ip-regex 1.0.3
1.2099.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2100 jsreport-office 1.2.0 1.2.0
1.2100.1 Available under license :
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2101 de.huxhorn.lilith.data.logging.protobuf
0.9.44

1.2101.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner
specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2102 vary 1.1.2 1.1.2
1.2102.1 Available under license:
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2103 private 0.1.8 0.1.8

1.2103.1 Available under license:

Copyright (c) 2014 Ben Newman <bn@cs.stanford.edu>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2104 aws4 1.8.0 1.8.0

1.2104.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2105 mime-db 1.35.0 1.35.0

1.2105.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
1.2106 pcre 2.48.2-0ubuntu4.4
1.2106.1 Available under license:

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

PCRE LICENCE

--------

PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself. The data in the testdata directory is not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions, and a just-in-time compiler that can be used to optimize pattern matching. These are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS

-----------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2020 University of Cambridge
All rights reserved.

PCRE JUST-IN-TIME COMPILATION SUPPORT

-----------------------------

Written by: Zoltan Herczeg
STACK-LESS JUST-IN-TIME COMPILER
-----------------------------

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2009-2020 Zoltan Herczeg
All rights reserved.

THE C++ WRAPPER FUNCTIONS
--------------------------

Contributed by: Google Inc.

Copyright (c) 2007-2012, Google Inc.
All rights reserved.

THE "BSD" LICENCE
-----------------

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2107 hbase-mapreduce 2.0.0-alpha4
1.2107.1 Available under license :

Apache HBase - MapReduce
Copyright 2007-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2108 regjsgen 0.2.0 0.2.0

1.2108.1 Available under license:

Copyright 2014 Benjamin Tan <demoneaux@gmail.com> (https://d10.github.io/)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2109 strip-json-comments 2.0.1

1.2109.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2110 yargs 15.4.1 15.4.1

1.2110.1 Available under license:
MIT License

Copyright 2010 James Halliday (mail@substack.net); Modified work Copyright 2014 Contributors (ben@npmjs.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2111 asm 3.3.1

1.2111.1 Available under license:
No license file was found, but licenses were detected in source scan.

/ ***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/MethodNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/SmallSet.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/ClassReader.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/InsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/ClassWriter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/MethodInsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/TableSwitchInsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/RemappingSignatureAdapter.java
Open Source Used In DNAC Platform Arica 10341

* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/MethodAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/FieldVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Item.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/LocalVariableNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/LabelNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Frame.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/inliner/InlinerNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/VarInsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifierAbstractVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/AnalyzerException.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/StaticInitMerger.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Attribute.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree.Handler.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Type.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/InsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/SourceInterpreter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Label.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/Opcodes.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/signature/SignatureReader.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/SimpleRemapper.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifierAnnotationVisitor.java
Open Source Used In DNAC Platform Arica  10342

* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/TraceMethodVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/ClassAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/RemappingClassAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifierFieldVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/Remapper.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/CheckMethodAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/CodeGenEvaluator.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/Subroutine.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/SourceValue.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/CheckClassAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/ClassVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/CheckSignatureAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/TraceAnnotationVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifierClassVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/FieldNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/CheckAnnotationAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/LocalVariablesSorter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/RemappingAnnotationAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/Frame.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/BasicVerifier.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/GeneratorAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/TraceFieldVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/FieldWriter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifierMethodVisitor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/InnerClassNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/LineNumberNode.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/RemappingMethodAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/InsnList.java

No license file was found, but licenses were detected in source scan.

/**
 * ASM XML Adapter
 * Copyright (c) 2004, Eugene Kuleshov
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/SAXCodeAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/Processor.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/SAXClassAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/SAXAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/ASMContentHandler.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/SAXAnnotationAdapter.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/SAXFieldAdapter.java

No license file was found, but licenses were detected in source scan.

* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/xml/package.html

No license file was found, but licenses were detected in source scan.

2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/commons/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/attrs/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/tree/analysis/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/package.html
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/signature/package.html

No license file was found, but licenses were detected in source scan.

/***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3.3.1-sources-6-
  jar/org/objectweb/asm/commons/InstructionAdapter.java
No license file was found, but licenses were detected in source scan.

/**
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF

---

Open Source Used In DNAC Platform Arica 10347
ASM XML Adapter
Copyright (c) 2004, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/ASMifiable.java
* /opt/cola/permits/1135935623_1613639042.9/0/asm-all-3-3-1-sources-6-jar/org/objectweb/asm/util/Traceable.java
No license file was found, but licenses were detected in source scan.

1.2112 optionator 0.8.3 0.8.3
1.2112.1 Available under license:
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2113 ansi-regex 5.0.1
1.2113.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2114 kind-of 3.2.2
1.2114.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2115 commons-beanutils 1.8.0

1.2115.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons BeanUtils
Copyright 2000-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.2116 tsconfig-paths 3.12.0
1.2116.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Jonas Kello

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2117 calcite-elasticsearch 1.21.0-drill-r2

1.2117.1 Available under license:

# Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to you under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and limitations under the License.
#
# Database of contributors to Apache Calcite.
# Pages such as developer.md use this data.
# List must be sorted by first name, last name.
#
- name: Alan Gates
  emeritus: 2018/05/04
apacheId: gates
githubId: alanfgates
org: Hortonworks
role: PMC
- name: Aman Sinha
  apacheId: amansinha
githubId: amansinha100
  org: MapR
  role: PMC
- name: Andrei Sereda
  apacheId: sereda
githubId: asereda-gs
  org: 
  role: Committer
- name: Ashutosh Chauhan
  apacheId: hashutosh
githubId: ashutoshc
  org: Hortonworks
  role: PMC
- name: Chris Baynes
  apacheId: cbaynes
githubId: chris-baynes
  org: Contiamo
  role: Committer
- name: Christian Beikov
  apacheId: cbeikov
  githubId: beikov
  org: Blazebit
  role: Committer
- name: Chunwei Lei
  apacheId: chunwei
githubId: chunweilei
  org: Alibaba
  role: Committer
- name: Danny Chan
  apacheId: danny0405
githubId: danny0405
  org: Alibaba
  role: Committer
- name: Edmon Begoli
  apacheId: ebegoli
  githubId: ebegoli
  org: Oak Ridge National Laboratory
  role: Committer
- name: Francis Chuang
  apacheId: francischuang
githubId: F21
  org: Boostport
role: PMC Chair
- name: Gian Merlino
  apacheId: gian
  githubId: gianm
  org: Imply
role: Committer
- name: Haisheng Yuan
  apacheId: hyuan
  githubId: hsyuan
  org: Alibaba
role: Committer
- name: Hongze Zhang
  apacheId: hongze
  githubId: zhtheplayer
  org: Tencent
role: Committer
- name: James R. Taylor
  apacheId: jamestaylor
  githubId: JamesRTaylor
  org: Salesforce
role: PMC
- name: Jacques Nadeau
  apacheId: jacques
  githubId: jacques-n
  org: Dremio
role: PMC
- name: Jess Balint
  apacheId: jbalint
  githubId: jbalint
  org: Stardog
role: Committer
- name: Jess Camacho Rodrguez
  apacheId: jcamacho
  githubId: jcamachor
  org: Hortonworks
role: PMC
- name: Jinfeng Ni
  apacheId: jni
  githubId: jinfengni
  org: MapR
role: PMC
- name: John Pullokkaran
  apacheId: jpullokk
  githubId: jpullokkaran
  org: Hortonworks
role: PMC
- name: Josh Elser
  apacheId: elserj
githubId: joshelser
org: Hortonworks
role: PMC
- name: Julian Hyde
  apacheId: jhyde
  githubId: julianhyde
  org: Looker
  role: PMC
  homepage: http://people.apache.org/~jhyde
- name: Kevin Liew
  apacheId: kliew
  githubId: kliewkliew
  org:
  role: Committer
- name: Kevin Risden
  apacheId: krisden
  githubId: risdenk
  org:
  role: PMC
- name: Laurent Goujon
  apacheId: laurent
  githubId: laurentgo
  org: Dremio
  role: PMC
- name: Maryann Xue
  apacheId: maryannxue
  githubId: maryannxue
  org: Intel
  role: PMC
- name: Michael Mior
  apacheId: mmior
  githubId: michaelmior
  org: Rochester Institute of Technology
  role: PMC
  homepage: https://michael.mior.ca/
- name: Milinda Pathirage
  apacheId: milinda
  githubId: milinda
  org: Indiana University
  role: Committer
  homepage: http://milinda.pathirage.org/
- name: MinJi Kim
  apacheId: minji
  githubId: minji-kim
  org: Oracle
  role: Committer
- name: Nick Dimiduk
  apacheId: ndimiduk
- name: Nishant Bangarwa
  apacheId: nishant
  githubId: nishantmonu51
  org: Hortonworks
  role: Committer

- name: Ruben Quesada Lopez
  apacheId: rubenql
  githubId: rubenada
  org: TIBCO
  role: Committer

- name: Sergey Nuyanzin
  apacheId: smuyanzin
  githubId: smuyanzin
  org: EPAM
  role: Committer

- name: Shuyi Chen
  apacheId: shuyichen
  githubId: suez1224
  org: Uber
  role: Committer

- name: Slim Bouguerra
  apacheId: bslim
  githubId: b-slim
  org: Hortonworks
  role: Committer

- name: Stamatis Zampetakis
  apacheId: zabetak
  githubId: zabetak
  org: TIBCO
  role: PMC

- name: Steven Noels
  apacheId: stevenn
  githubId: stevenn
  org: NGData
  role: PMC

- name: Ted Dunning
  apacheId: tdunning
  githubId: tdunning
  org: MapR
  role: PMC
  avatar: https://mapr.com/blog/author/ted-dunning/assets/tdunning-panama.jpg

- name: Vladimir Sitnikov
  apacheId: vladimirsitnikov
  githubId: vlsi
org: NetCracker
role: PMC
- name: Volodymyr Vysotskyi
  apacheId: volodymyr
  githubId: vvysotskyi
  org:
  role: PMC
- name: Zhen Wang
  apacheId: zhenw
  githubId: zinking
  org:
  role: Committer
- name: Zhiqiang He
  apacheId: zhiqianghe
  githubId: Zhiqiang-He
  org: Huawei
  role: Committer
- name: Zhiwei Peng
  apacheId: zhiwei
  githubId: pengzhiwei2018
  org: Ant Financial Group
  role: Committer
- name: Zoltan Haindrich
  apacheId: kgyrtkirk
  githubId: kgyrtkirk
  org: Hortonworks
  role: Committer

# End contributors.yml

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable 
copyright license to reproduce, prepare Derivative Works of, 
publicly display, publicly perform, sublicense, and distribute the 
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of 
this License, each Contributor hereby grants to You a perpetual, 
worldwide, non-exclusive, no-charge, royalty-free, irrevocable 
(except as stated in this section) patent license to make, have made, 
use, offer to sell, sell, import, and otherwise transfer the Work, 
where such license applies only to those patent claims licensable 
by such Contributor that are necessarily infringed by their 
Contribution(s) alone or by combination of their Contribution(s) 
with the Work to which such Contribution(s) was submitted. If You 
institute patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Work 
or a Contribution incorporated within the Work constitutes direct 
or contributory patent infringement, then any patent licenses 
granted to You under this License for that Work shall terminate 
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the 
Work or Derivative Works thereof in any medium, with or without 
modifications, and in Source or Object form, provided that You 
meet the following conditions:

(a) You must give any other recipients of the Work or 
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices 
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works 
that You distribute, all copyright, patent, trademark, and 
attribution notices from the Source form of the Work, 
excluding those notices that do not pertain to any part of 
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its 
distribution, then any Derivative Works that You distribute must 
include a readable copy of the attribution notices contained 
within such NOTICE file, excluding those notices that do not 
pertain to any part of the Derivative Works, in at least one 
of the following places: within a NOTICE text file distributed 
as part of the Derivative Works; within the Source form or 
documentation, if provided along with the Derivative Works; or, 
within a display generated by the Derivative Works, if and 
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------

APACHE CALCITE SUBCOMPONENTS:

The Apache Calcite project contains subcomponents with separate copyright
notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

-----------------------------------------------------------------------
The MIT License
-----------------------------------------------------------------------

The Apache Calcite project bundles the following files under the MIT License:

- site
  Parts of the web site generated by Jekyll (http://jekyllrb.com/)
  Copyright (c) 2008-2015 Tom Preston-Werner
- site/_sass/_font-awesome.scss
  Font-awesome css files v4.1.0 (http://fortawesome.github.io/Font-Awesome/)
  Copyright (c) 2013 Dave Gandy
- site/_sass/_normalize.scss
  normalize.css v3.0.2 | git.io/normalize
  Copyright (c) Nicolas Gallagher and Jonathan Neal
- site/_sass/_gridism.scss
  Gridism: A simple, responsive, and handy CSS grid by @cobyism
  https://github.com/cobyism/gridism
  Copyright (c) 2013 Coby Chapple
- site/js/html5shiv.min.js
  HTML5 Shiv 3.7.2 | @afarkas @jdalton @jon_neal @rem
- site/js/respond.min.js
  Respond.js v1.4.2: min/max-width media query polyfill
  Copyright 2013 Scott Jehl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------
The Open Font License
------------------------------------------------------------------------

The Apache Calcite project bundles the following fonts under the
SIL Open Font License (OFL) - http://scripts.sil.org/OFL/

- site/fonts/fontawesome-webfont.*
  Font-awesome font files v4.0.3 (http://fortawesome.github.io/Font-Awesome/)
Apache Calcite
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.

The web site includes files generated by Jekyll.

1.2118 cli-boxes 2.2.1
1.2118.1 Available under license :
  MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2119 urijs 1.19.6 1.19.6
1.2119.1 Available under license:
The MIT License (MIT)

Copyright (c) 2011 Rodney Rehm

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2120 babel-plugin-syntax-nullish-coalescing-operator 7.8.3

1.2120.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
1.2121 p-locate 3.0.0

1.2121.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2122 combined-stream 1.0.6 1.0.6

1.2122.1 Available under license :
Copyright (c) 2011 Debuggable Limited <felix@debuggable.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2123 websocket-common 9.3.27.v20190418

1.2124 swagger2openapi 7.0.3 7.0.3
1.2124.1 Available under license:
   BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2125 shelljs 0.7.8
**1.2125.1 Available under license:**
Copyright (c) 2012, Artur Adib <arturadib@gmail.com>
All rights reserved.

You may use this project under the terms of the New BSD license as follows:

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of Artur Adib nor the
  names of the contributors may be used to endorse or promote products
  derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL ARTUR ADIB BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**1.2126 @babel/helper-hoist-variables 7.16.0**

**1.2126.1 Available under license:**
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2127 lodash.reduce 4.5.0 4.5.0

1.2127.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

### 1.2128 detect-indent 4.0.0

**1.2128.1 Available under license :**
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.2129 classnames 2.2.6 2.2.6

**1.2129.1 Available under license :**
The MIT License (MIT)

Copyright (c) 2017 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2130 okhttp 4.9.1
1.2130.1 Available under license:
Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/
Mozilla Public License
Version 2.0
1. Definitions
1.1. “Contributor”
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. “Contributor Version”
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s
Contribution.

1.3. “Contribution”
means Covered Software of a particular Contributor.

1.4. “Covered Software”
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form
of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. “Incompatible With Secondary Licenses”
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also
under the terms of a Secondary License.
1.6. “Executable Form”
means any form of the work other than Source Code Form.

1.7. “Larger Work”
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. “License”
means this document.

1.9. “Licensable”
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. “Modifications”
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. “Secondary License”
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form”
means the form of the work preferred for making modifications.

1.14. “You” (or “Your”)
means an individual or a legal entity exercising rights under this License. For legal entities, “You” includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions
2.1. Grants
Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an
unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer
either its Contributions or its Contributor Version.

2.2. Effective Date
The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the
date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope
The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses
will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section
2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the
combination of its Contributions with other software (except as part of its Contributor Version); or

under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may
be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses
No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a
subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under
the terms of Section 3.3).

2.5. Representation
Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has
sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use
This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair
dealing, or other equivalents.

2.7. Conditions
Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities
3.1. Distribution of Source Form
All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which
You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of
the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You
may not attempt to alter or restrict the recipients’ rights in the Source Code Form.

3.2. Distribution of Executable Form
If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You
must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable
means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms,
provided that the license for the Executable Form does not attempt to limit or alter the recipients’ rights in the
Source Code Form under this License.

3.3. Distribution of a Larger Work
You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the
requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software
with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With
Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of
such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the
Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices
You may not remove or alter the substance of any license notices (including copyright notices, patent notices,
disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software,
except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms
You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more
recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any
Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is
offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such
Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered
Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the
maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed
in a text file included with all distributions of the Covered Software under this License. Except to the extent
prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to
be able to understand it.

5. Termination
5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms.
However, if You become compliant, then the rights granted under this License from a particular Contributor are
reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b)
on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior
to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are
reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means,
this is the first time You have received notice of non-compliance with this License from such Contributor, and You
become compliant prior to 30 days after Your receipt of the notice.
5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty
Covered Software is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability
Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation
Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous
This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License
10.1. New Versions
Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions
You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions
If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses
If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - “Incompatible With Secondary Licenses” Notice
This Source Code Form is “Incompatible With Secondary Licenses”, as defined by the Mozilla Public License, v. 2.0.
/*
 * Copyright (C) 2014 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

**1.2131 abbrev 1.0.9**

**1.2131.1 Available under license:**
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2132 jackson 2.4.0

1.2132.1 Available under license:

== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0, ==
== in this case for the Apache Camel distribution. ==

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Please read the different LICENSE files present in the licenses directory of this distribution.

Camel :: Jackson
Copyright 2007-2010 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2133 ajv-keywords 3.2.0 3.2.0
1.2133.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2134 zstd-jni 1.4.9-1

1.2134.1 Available under license:

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.github.luben.zstd_jni
Bnd-LastModified: 1615191527618
Bundle-Description: JNI bindings for Zstd native library that provides fast and high compression lossless algorithm for Java and all JVM languages.
Bundle-License: https://opensource.org/licenses/BSD-2-Clause;description=BSD 2-Clause License
Bundle-ManifestVersion: 2
Bundle-Name: zstd-jni
Bundle-NativeCode: aix/ppc64/libzstd-jni.so;osname=AIX;processor=ppc64 , darwin/x86_64/libzstd-jni.dylib;osname=MacOS;osname=MacOSX;processor=x86_64, darwin/aarch64/libzstd-jni.dylib;osname=MacOS;osname=MacOSX;processor=aarch64, freebsd/amd64/libzstd-jni.so;osname=FreeBSD;processor=amd64, freebsd/i386/libzstd-jni.so;osname=FreeBSD;processor=i386, linux/aarch64/libzstd-jni.so;osname=Linux;processor=aarch64, linux/amd64/libzstd-jni.so;osname=Linux;processor=amd64, linux/arm/libzstd-jni.so;osname=Linux;processor=arm, linux/i386/libzstd-jni.so;osname=Linux;processor=i386, linux/mips64/libzstd-jni.so;osname=Linux;processor=mips64
1.2135 format-util 1.0.5 1.0.5

1.2135.1 Available under license:

Copyright 2015 Freeform Systems and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

NO WARRANTY OR LIABILITY

YOU EXPRESSLY AGREE THAT DOWNLOADING THE SOFTWARE AND ANY USE OF THE SOFTWARE IS AT YOUR OWN RISK. NO WARRANTY, REPRESENTATION, CONDITION, UNDERTAKING OR TERM - EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE - INCLUDING BUT NOT LIMITED TO THE CONDITION, QUALITY, DURABILITY, PERFORMANCE, ACCURACY, STABILITY, RELIABILITY, NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE OR USE OF THE SOFTWARE IS GIVEN OR ASSUMED BY FREEFORM SYSTEMS LTD. ALL SUCH WARRANTIES, REPRESENTATIONS, CONDITIONS, UNDERTAKINGS AND TERMS ARE HEREBY EXCLUDED. FREEFORM SYSTEMS LTD MAKES NO WARRANTY THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT IT WILL BE UNINTERRUPTED, TIMELY, SECURE, OR ERROR FREE; IN NO EVENT SHALL FREEFORM SYSTEMS LTD BE LIABLE TO ANY PARTY FOR ANY DAMAGES INCLUDING WITHOUT LIMITATION, ANY DIRECT, INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF PROGRAMS OR INFORMATION, LOSS OF PROFITS AND SAVINGS AND THE LIKE), OR ANY OTHER DAMAGES ARISING - IN ANY WAY, SHAPE OR FORM - OUT OF THE AVAILABILITY, USE, RELIANCE ON, INABILITY TO UTILIZE OR IMPROPER USE OF THE SOFTWARE EVEN IF FREEFORM SYSTEMS LTD SHALL HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, AND REGARDLESS OF THE FORM OF ACTION, WHETHER IN CONTRACT, TORT, OR OTHERWISE. BECAUSE SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, THE ABOVE EXCLUSIONS OF INCIDENTAL AND CONSEQUENTIAL DAMAGES MAY NOT APPLY TO YOU.

1.2136 ws 6.1.4 6.1.4

1.2136.1 Available under license :

The MIT License (MIT)

Copyright (c) 2011 Einar Otto Stangvik <einaros@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2137 regenerator-runtime 0.10.5
1.2137.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2014, Facebook, Inc.
 * All rights reserved.
 *
 * This source code is licensed under the BSD-style license found in the
 * https://raw.githubusercontent.com/facebook/regenerator/master/LICENSE file. An
 * additional grant of patent rights can be found in the PATENTS file in
 * the same directory.
 */

Found in path(s):
* /opt/cola/permits/1110631590_1606842066.7/0/regenerator-runtime-0-10-5-4-tgz/package/runtime.js

1.2138 jffi 1.2.15
1.2138.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.
If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

**1.2139 babylon 6.18.0 6.18.0**

**1.2139.1 Available under license:**

```
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

**1.2140 libxi 1.7.6-1**

**1.2140.1 Available under license:**

```
Copyright 1989, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
```
OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be
used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization from The Open Group.

Copyright 1989 by Hewlett-Packard Company, Palo Alto, California.

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Hewlett-Packard not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

HEWLETT-PACKARD DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO
EVENT SHALL HEWLETT-PACKARD BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

Copyright 2008 Peter Hutterer

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation.

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHOR BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the author shall not be
used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization from the author.

Copyright 2009 Red Hat, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2141 install 0.12.2 0.12.2

1.2141.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Benjamin Newman <ben@benjamn.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2142 commons-io 2.7
1.2142.1 Available under license:

Apache Commons IO
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2143 esprima 4.0.1
1.2143.1 Available under license :

Copyright JS Foundation and other contributors, https://js.foundation/

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.2144 mylib 0.0.0

1.2144.1 Available under license:

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy
deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.


1.2145.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
1.2146 yeast 0.1.2 0.1.2
1.2146.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2147 de.huxhorn.lilith.data.eventsource
0.9.44
1.2147.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser
General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the
Library.

/*
 * Lilith - a log event viewer.
 * Copyright (C) 2007-2011 Joern Huxhorn
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU Lesser General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/*
 * Copyright 2007-2011 Joern Huxhorn
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

1.2148 strip-ansi 6.0.0
1.2148.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2149 zookeeper-jute 3.5.7
1.2149.1 Available under license :

Apache ZooKeeper - Jute
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2150 pinkie 1.0.0 1.0.0

1.2150.1 Available under license:
The MIT License (MIT)

Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2151 @babel/plugin-proposal-class-properties 7.16.0

1.2151.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2152 is-buffer 2.0.4 2.0.4
1.2152.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2153 wrap-ansi 6.2.0
1.2153.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.2154 dont-sniff-mimetype 1.0.0 1.0.0

1.2154.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014, 2015 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2155 pdfjs-dist 2.0.305 2.0.305

1.2155.1 Available under license:

%%Copyright: -----------------------------------------------------------
%%Copyright: Copyright 1990-2009 Adobe Systems Incorporated.
%%Copyright: All rights reserved.
%%Copyright: Redistribution and use in source and binary forms, with or
%%Copyright: without modification, are permitted provided that the
%%Copyright: following conditions are met:
%%Copyright: Redistributions of source code must retain the above
%%Copyright: copyright notice, this list of conditions and the following
%%Copyright: disclaimer.
%%Copyright: Redistributions in binary form must reproduce the above
%%Copyright: copyright notice, this list of conditions and the following
%%Copyright: disclaimer in the documentation and/or other materials
%%Copyright: provided with the distribution.
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, 
use, offer to sell, sell, import, and otherwise transfer the Work, 
where such license applies only to those patent claims licensable 
by such Contributor that are necessarily infringed by their 
Contribution(s) alone or by combination of their Contribution(s) 
with the Work to which such Contribution(s) was submitted. If You 
institute patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Work 
or a Contribution incorporated within the Work constitutes direct 
or contributory patent infringement, then any patent licenses 
granted to You under this License for that Work shall terminate 
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the 
Work or Derivative Works thereof in any medium, with or without 
modifications, and in Source or Object form, provided that You 
meet the following conditions:

(a) You must give any other recipients of the Work or 
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices 
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works 
    that You distribute, all copyright, patent, trademark, and 
    attribution notices from the Source form of the Work, 
    excluding those notices that do not pertain to any part of 
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its 
    distribution, then any Derivative Works that You distribute must 
    include a readable copy of the attribution notices contained 
    within such NOTICE file, excluding those notices that do not 
    pertain to any part of the Derivative Works, in at least one 
    of the following places: within a NOTICE text file distributed 
    as part of the Derivative Works; within the Source form or 
    documentation, if provided along with the Derivative Works; or, 
    within a display generated by the Derivative Works, if and 
    wherever such third-party notices normally appear. The contents 
    of the NOTICE file are for informational purposes only and 
    do not modify the License. You may add Your own attribution 
    notices within Derivative Works that You distribute, alongside 
    or as an addendum to the NOTICE text from the Work, provided 
    that such additional attribution notices cannot be construed 
    as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2156 libdrm 2.4.91 2~16.04.1

1.2157 follow-redirects 1.5.10 1.5.10
1.2157.1 Available under license:
Copyright 2014-present Olivier Lalonde <olalonde@gmail.com>, James Talmage <james@talmage.io>, Ruben Verborgh

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR
IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2158 drill-format-hdf5 1.19.0
1.2158.1 Available under license:
Drill : Contrib : Format : HDF5
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2159 libvpx 1.7.0-1367-gac3eccdc2
1.2159.1 Available under license:

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2005-2012 x264 project

Authors: Loren Merritt <lorenm@u.washington.edu>
Anton Mitrofanov <BugMaster@narod.ru>
Jason Garrett-Glaser <darkshikari@gmail.com>
Henrik Gramner <hengar-6@student.itu.se>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 2010, The WebM Project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google, nor the WebM Project, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2011 The LibYuv Project Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# This file contains a list of people who've made non-trivial
# contribution to the Google C++ Testing Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlere@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@google.com>
Ji Sigursson <joi@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@google.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@google.com>
Patrick Riley <pfr@google.com>
Peter Kaminski <piotr@k@gmail.com>
Preston Jackson <preston.a.jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@google.com>
Russ Rufer <russ@pentad.com>
Sean Mcafee <seefacm@gmail.com>
Sigurur sgeirsson <sigg@google.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@google.com>
Zhanyong Wan <wan@google.com>

1.2160 calcite-example-function 1.21.0
1.2160.1 Available under license:

Calcite Example Function
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2161 decompress-response 3.3.0

1.2161.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2162 hive-metastore 3.1.2

1.2162.1 Available under license:

Hive Metastore
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing. Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2163 wrap-ansi 7.0.0 7.0.0
1.2163.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTicular PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2164 function-bind 1.1.1 1.1.1
1.2164.1 Available under license :
Copyright (c) 2013 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2165 findup-sync 3.0.0 3.0.0

1.2165.1 Available under license :

The MIT License (MIT)

Copyright (c) 2013-2018 Ben Alman <cowboy@rj3.net>, Blaine Bublitz <blaine.bublitz@gmail.com>, and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2166 hive-shims-common 3.1.2

1.2166.1 Available under license :

Hive Shims Common
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.  
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.2167 yargs-parser 18.1.3 18.1.3
1.2167.1 Available under license :

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software 
for any purpose with or without fee is hereby granted, provided 
that the above copyright notice and this permission notice 
appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES 
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES 
OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE 
LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES 
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, 
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, 
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2168 hive-standalone-metastore 3.1.2
1.2168.1 Available under license :

Hive Standalone Metastore
Copyright 2019 The Apache Software Foundation

This product includes software developed at 
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.2169 request-promise-core 1.1.4 1.1.4

1.2169.1 Available under license:
ISC License

Copyright (c) 2020, Nicolai Kamenzky and contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2170 javax.servlet.jsp 2.3.2

1.2170.1 Available under license:

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License

Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in '# Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. # The contents of this file are subject to the terms of either the GNU # General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You # may not use this file except in compliance with the License. You can # Oracle designates this particular file as subject to the "Classpath" # exception as provided by Oracle in the GPL Version 2 section of the License # Licensed under the Apache License, Version 2.0 (the "License"); # you may not use this file except in compliance with the License. # You may obtain a copy of the License at # http://www.apache.org/licenses/LICENSE-2.0 # distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'
Found license 'General Public License 2.0' in 'Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License <bottom> Copyright (c) 1999-2012 Oracle and/or its affiliates. All Rights Reserved. Use is subject to license terms. </bottom>’

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2011 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License <bottom> Copyright (c) 1999-2012 Oracle and/or its affiliates. All Rights Reserved. Use is subject to license terms. </bottom>’

Found license 'General Public License 2.0' in '* Copyright (c) 1997-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'
Open Source Used In DNAC Platform Arica 10448

Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS,

Found license 'General Public License 2.0' in * Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License' Found license 'General Public License 2.0' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License' Found license 'General Public License 2.0' in * Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS,'

1.2171 javax-inject 1

1.2171.1 Available under license :

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

/*
 * Copyright (C) 2009 The JSR-330 Expert Group
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
1.2172 hive-exec 3.1.2
1.2172.1 Available under license:

Hive Query Language
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

---

Open Source Used In DNAC Platform Arica 10454
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2173 yargs-parser 13.1.2

1.2173.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2174 setprototypeof 1.1.0 1.1.0

1.2174.1 Available under license:

Copyright (c) 2015, Wes Todd

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2175 debug 3.2.7

1.2175.1 Available under license:

(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2176 ansi-align 3.0.0

1.2176.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright notice
and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF
THIS SOFTWARE.

1.2177 commons-pool 2.4.3

1.2177.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS POOL DERIVATIVE WORKS:

The LinkedBlockingDeque implementation is based on an implementation written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at

http://creativecommons.org/licenses/publicdomain

Apache Commons Pool
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

---

This project incorporates portions of the 'Protocol Buffers' project available
under a '3-clause BSD' license.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

Apache HBase Patched and Relocated (Shaded) Protobuf
Copyright 2017-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

---

// Protocol Buffers - Google's data interchange format
// Copyright 2008 Google Inc. All rights reserved.
// https://developers.google.com/protocol-buffers/
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// distribution.
// * Neither the name of Google Inc. nor the names of its
// contributors may be used to endorse or promote products derived from
// this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.2179 require-main-filename 2.0.0 2.0.0

1.2179.1 Available under license :

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2180 stream-combiner 0.0.4 0.0.4

1.2180.1 Available under license :

Copyright (c) 2012 'Dominic Tarr'

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2181 log4j-slf4j-impl 2.17.1

1.2181.1 Available under license:

Apache Log4j SLF4J Binding
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2182 es6-promisify 5.0.0 5.0.0

1.2182.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.2183 semver 7.0.0

1.2183.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2184 is-regex 1.1.4

1.2184.1 Available under license:
MIT

1.2185 cors 2.8.5 2.8.5

1.2185.1 Available under license:
(The MIT License)

Copyright (c) 2013 Troy Goode <troygoode@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2186 jetty-io 9.4.34.v20201102
1.2186.1 Available under license:

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates'
from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.
"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or
Open Source Used In DNAC Platform Arica  10479

conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and
consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.
Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications
are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay
The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL/GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.2187 jsrender 1.0.4 1.0.4

1.2187.1 Available under license:

Copyright (c) 2019 Boris Moore https://github.com/BorisMoore/jsrender

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2188 type-of-is 3.5.1 3.5.1

1.2188.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2189 which 1.3.1
1.2189.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2190 kind-of 6.0.3 6.0.3
1.2190.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2191 p-map 3.0.0
1.2191.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2192 hdrhistogram 2.1.12
1.2192.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 */
Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/WriterReaderPhaser.java
No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * @author Gil Tene
 */

Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoublePercentileIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ZigZagEncoding.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AllValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/Base64Helper.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/IntCountsHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleLinearIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/PackedConcurrentDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/SynchronizedHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/logarithmic/LogarithmicIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/Version.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/RecordedValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/LinearIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/PackedDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AtomicHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/HistogramIterationValue.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/Recorder.java
No license file was found, but licenses were detected in source scan.

/*
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 * @author Gil Tene
 */
1.2193 open-csv 5.4

1.2193.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
   Copyright 2015 Bytecode Pty Ltd.

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/
/**
 * A very simple CSV writer released under a commercial-friendly license.
 *
 * @author Glen Smith
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/CSVWriter.java
No license file was found, but licenses were detected in source scan.

<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/META-INF/maven/com.opencsv/opencsv/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2007 Kyle Miller.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/HeaderColumnNameMappingStrategy.java
No license file was found, but licenses were detected in source scan.

Open Source Used In DNAC Platform Arica 10484
/*
 * Copyright 2015 Scott Conway
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/ResultSetColumnNameHelperService.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2007,2010 Kyle Miller.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/HeaderColumnNameTranslateMappingStrategy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2005 Bytecode Pty Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/
/**
 * A very simple CSV parser released under a commercial-friendly license.
 * This just implements splitting a single line into fields.</p>
 *
 * The purpose of the CSVParser is to take a single string and parse it into
 * its elements based on the delimiter, quote and escape characters.</p>
 *
 * The CSVParser has grown organically based on user requests and does not truly match
 * any current requirements (though it can be configured to match or come close). There
 * is no plans to change this as it will break existing requirements. Consider using
 * the RFC4180Parser for less configurability but closer match to the RFC4180 requirements.</p>
 *
 * @author Glen Smith
 * @author Rainer Pruy
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/CSVParser.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 Andrew Rucker Jones.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvConverter.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-
jar/com/opencsv/bean/BeanFieldSingleValue.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvBindAndJoinByPosition.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/AccumulateCsvResults.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/FieldMapByNameEntry.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/FieldMapByName.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/ProcessCsvBean.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/IntolerantThreadPoolExecutor.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/ProcessCsvLine.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvBindAndJoinByName.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvBindAndSplitByPosition.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/ProcessCsvResult.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/OrderedObject.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/BeanFieldJoin.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvBindAndSplitByName.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/AbstractCsvConverter.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/BeanFieldSplit.java

No license file was found, but licenses were detected in source scan.

/*
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/
/**
 * A very simple CSV reader released under a commercial-friendly license.
 *
 * @author Glen Smith
/*

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/CSVReader.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2018 Andrew Rucker Jones.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/PositionToBeanField.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/BeanFieldJoinStringIndex.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/ConverterNumber.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/comparator/LiteralComparator.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/AbstractMappingStrategy.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/RegexToBeanField.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/ComplexFieldMapEntry.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/FieldMapByPosition.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/AbstractFieldMap.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/FieldMapByPositionEntry.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/BeanFieldJoinIntegerIndex.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 Bytecode Pty Ltd.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0

 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/CSVParserWriter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 Andrew Rucker Jones.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0

 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/package-info.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/util/OpencsvUtils.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/ConverterDate.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/HeaderIndex.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/AbstractFieldMapEntry.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvNumber.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 Mark Rogers.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvToBeanFilter.java
No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/convertLanguageToBoolean.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/convertLanguageToBoolean_en.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/opencsv_en.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/opencsv.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/opencsv_pt_BR.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/convertLanguageToBoolean_de.properties
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/opencsv_de.properties
No license file was found, but licenses were detected in source scan.

/*
Copyright 2005 Bytecode Pty Ltd.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/opencsv-5-4-sources-jar/com/opencsv/ResultSetHelperService.java
* /opt/cola/permits/1258286328_1643052526.49/opencsv-5-4-sources-jar/com/opencsv/ResultSetHelper.java
* /opt/cola/permits/1258286328_1643052526.49/opencsv-5-4-sources-jar/com/opencsv/CSVParserBuilder.java
* /opt/cola/permits/1258286328_1643052526.49/opencsv-5-4-sources-jar/com/opencsv/CSVReaderBuilder.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 arjones.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/opencsv-5-4-sources-jar/com/opencsv/bean/concurrent/package-info.java

No license file was found, but licenses were detected in source scan.

/**
 * A very simple CSV parser for Java released under a commercial-friendly license.
 *
 * @see <a href="http://opencsv.sourceforge.net/">opencsv.sourceforge.net</a>
 */

/*
 * Copyright 2016 Andrew Rucker Jones.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/package-info.java

No license file was found, but licenses were detected in source scan.

/*
Copyright 2007 Kyle Miller.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/CsvToBean.java
* /opt/cola/permits/1258286328_1643052526.49/0/opencsv-5-4-sources-jar/com/opencsv/bean/MappingStrategy.java

1.2194 performance-now 2.1.0 2.1.0

1.2194.1 Available under license:

Copyright (c) 2013 Braveg1rl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2195 raml-to-swagger 1.1.0 1.1.0
1.2195.1 Available under license :

Copyright (c) 2015,
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of raml-to-swagger nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2196 mimic-fn 2.1.0 2.1.0
1.2196.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2197 parseuri 0.0.6 0.0.6

1.2197.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Gal Koren

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2198 babel-plugin-transform-runtime 6.23.0
1.2198.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2199 jcommander 1.78
1.2199.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2010 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/IUsageFormatter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/UnixStyleUsageFormatter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/JCommander.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/DefaultUsageFormatter.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright (C) 2010 the original author or authors.
* See the notice.md file distributed with this work for additional
* information regarding copyright ownership.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/IStringConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/URLConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/internal/Lists.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/ParameterException.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/internal/DefaultConverterFactory.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/IntegerConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2.jar/com/beust/jcommander/converters/DefaultConverterFactory.java
jar/com/beust/jcommander/converters/FileConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/StringConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/IDStringConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/internal/Sets.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/LongConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/internal/Maps.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/InetAddressConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/Parameters.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/ParameterDescription.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/URIConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/BaseConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/CharArrayConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/PathConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/ParametersDelegate.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/ResourceBundle.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/defaultprovider/PropertyFileDefaultProvider.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/IDefaultProvider.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/DoubleConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/BigDecimalConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/FloatConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/MissingCommandException.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/NoConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/converters/BooleanConverter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
  jar/com/beust/jcommander/Parameter.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-
jar/com/beust/jcommander/converters/ISO8601DateConverter.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2011 the original author or authors.
 * See the notice.md file distributed with this work for additional
 * information regarding copyright ownership.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/validators/NoValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/validators/PositiveInteger.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/validators/NoValueValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/IParameterValidator.java
* /opt/cola/permits/1135840359_1613613123.36/0/jcommander-1-78-sources-2-jar/com/beust/jcommander/IParameterValidator2.java

1.2200 uuid 3.2.1 3.2.1
1.2200.1 Available under license :
The MIT License (MIT)

Copyright (c) 2010-2016 Robert Kieffer and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
1.2201 @babel/plugin-transform-modules-systemjs 7.16.0

1.2201.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2202 type 1.2.0

1.2202.1 Available under license:
No license file was found, but licenses were detected in source scan.

/!
* mocha
1.2203 drill-mongo-storage 1.19.0

1.2203.1 Available under license:

Drill : Contrib : Storage : MongoDB
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2204 babel-helper-optimise-call-expression

6.24.1 6.24.1

1.2204.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2205 for-own 1.0.0 1.0.0
1.2205.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2206.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2207 opentracing-noop 0.31.0
1.2207.1 Available under license:

No license file was found, but licenses were detected in source scan.

2018 The OpenTracing Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1000002513_1646171449.18/0/opentracing-noop-0-31-0-sources-jar/META-INF/maven/io.opentracing/opentracing-noop/pom.xml
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2016-2018 The OpenTracing Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except *
* in compliance with the License. You may obtain a copy of the License at *
* *
* http://www.apache.org/licenses/LICENSE-2.0 *
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License *
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express *
* or implied. See the License for the specific language governing permissions and limitations under *
* the License. *
*/
1.2208 got 2.4.0 2.4.0
1.2208.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2209 babel-preset-env 1.7.0
1.2209.1 Available under license:
MIT License

Copyright (c) 2016-2017 Babel

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2210 repeating 2.0.1

1.2210.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2211 babel-plugin-transform-es2015-spread

6.22.0
1.2211.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. See the
ECMA

CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.2212 jna 3.5.1

1.2212.1 Available under license:

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007 Wayne Meissner, All Rights Reserved
This library * is free software; you can redistribute it and/or modify it under the * terms of the GNU Lesser General Public License as published by the * Free Software Foundation; either version 2.1 of the License, or (at * your option) any later version. <p/> This library is distributed in * PARTICULAR PURPOSE. See the GNU Lesser General Public License for * more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007 Wayne Meissner, All Rights Reserved
* Copyright (c) 2007, 2008, 2009 Timothy Wall, All Rights Reserved * This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public * License as published by the Free Software Foundation; either * version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007 Timothy Wall, All Rights Reserved *
This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public License as published by the * Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007 Timothy Wall, All Rights Reserved *
This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public License as published by the * Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2012 Timothy Wall, All Rights Reserved *
This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2009 Timothy Wall, All Rights Reserved *
This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'
Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007-2008 Timothy Wall, All Rights Reserved * This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public * License as published by the Free Software Foundation; either * version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2011 Timothy Wall, All Rights Reserved * This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public * License as published by the Free Software Foundation; either * version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007 Wayne Meissner, All Rights Reserved * This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public * License as published by the Free Software Foundation; either * version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

Found license 'GNU Lesser General Public License' in /* Copyright (c) 2007-2012 Timothy Wall, All Rights Reserved * This library is free software; you can redistribute it and/or * modify it under the terms of the GNU Lesser General Public * License as published by the Free Software Foundation; either * version 2.1 of the License, or (at your option) any later version. * This library is distributed in the hope that it will be useful, * but WITHOUT ANY WARRANTY; without even the implied warranty of * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU * Lesser General Public License for more details.'

---

**1.2213 jsreport-resources 2.1.0 2.1.0**

**1.2213.1 Available under license:**

**GNU LESSER GENERAL PUBLIC LICENSE**  
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates  
the terms and conditions of version 3 of the GNU General Public  
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser  
General Public License, and the "GNU GPL" refers to version 3 of the GNU  
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL., with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user’s computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.
e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the Library.

1.2214 babel-template 6.26.0

1.2214.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2215 jsbn 1.1.0 1.1.0
1.2215.1 Available under license:

Licensing
---------

This software is covered under the following copyright:

/ *
* Copyright (c) 2003-2005 Tom Wu
* All Rights Reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
* 
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND,
* EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY
* WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
*
* IN NO EVENT SHALL TOM WU BE LIABLE FOR ANY SPECIAL, INCIDENTAL,
* INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER
* RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF
* THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
* 
* In addition, the following condition applies:
* 
* All redistributions must retain an intact copy of this copyright notice
* and disclaimer.
*/
Address all questions regarding this license to:

Tom Wu
tjw@cs.Stanford.EDU

1.2216 anymatch 3.1.2

1.2216.1 Available under license :
The ISC License

Copyright (c) 2019 Elan Shanker, Paul Miller (https://paulmillr.com)

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2217 find-up 3.0.0 3.0.0

1.2217.1 Available under license :
The MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2218 netty-handler 4.1.59.Final

1.2218.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/SslContextBuilder.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/OpenSslSessionTicketKey.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/ClientAuth.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/ApplicationProtocolAccessor.java

No license file was found, but licenses were detected in source scan.

/

* Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/OpenSslPrivateKeyMethod.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/KeyManagerFactoryWrapper.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/SslMasterKeyHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/address/package-info.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/X509KeyManagerWrapper.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/PseudoRandomFunction.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/address/DynamicAddressConnectHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/TrustManagerFactoryWrapper.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/SimpleKeyManagerFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/LazyX509Certificate.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ipfilter/IPFilterRule.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/OpenSslSessionContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/SslProvider.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/JdkSslSessionContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/JdkNpnSslSessionContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/JdkSslProvider.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/util/OpenJdkSelfSignedCertGenerator.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/util/SimpleTrustManagerFactory.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/PemReader.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/InsecureTrustManagerFactory.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/SupportedCipherSuiteFilter.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/JdkSslClientContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/OpenSslServerContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/SslUtils.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/JdkSslServerContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ipfilter/IPSubnetFilterRule.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/CipherSuiteConverter.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/OpenSslEngineMap.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/traffic/GlobalChannelTrafficCounter.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/ApplicationProtocolNegotiator.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ipfilter/package-info.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/ApplicationProtocolConfig.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/OpenSslContext.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/SniHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ss/Util/FingerprintTrustManagerFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/logging/LogLevel.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/stream/package-info.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/stream/ChunkedWriteHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/timeout/ReadTimeoutHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/logging/LoggingHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/stream/WriteTimeoutException.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/stream/ChunkedFile.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/timeout/IdleStateHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslCachingX509KeyManagerFactory.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslKeyMaterialProvider.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslPrivateKey.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslX509TrustManagerWrapper.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslX509KeyManagerFactory.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OpenSslSession.java

No license file was found, but licenses were detected in source scan.

* Copyright 2012 The Netty Project

* The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Open Source Used In DNAC Platform Arica 10544
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/traffic/TrafficCounter.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/traffic/GlobalTrafficShapingHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/traffic/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/traffic/AbstractTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/ocsp/package-info.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/Conscrypt.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/AbstractSniHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/JdkAlpnSslUtils.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/SslClientHelloHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/SslCompletionEvent.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/SniCompletionEvent.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/JdkAlpnSslEngine.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OcspClientHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/SslCloseCompletionEvent.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/OptionalSslHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-
  jar/io/netty/handler/ssl/ConscryptAlpnSslEngine.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/flow/FlowControlHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/EthernetPacket.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/IPPacket.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/UDPPacket.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/TCPPacket.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/PcapWriter.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/package-info.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/PcapHeaders.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/pcap/PcapWriteHandler.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2021 The Netty Project 
* 
* The Netty Project licenses this file to you under the Apache License, 
* version 2.0 (the "License"); you may not use this file except in compliance 
* with the License. You may obtain a copy of the License at: 
* 
* https://www.apache.org/licenses/LICENSE-2.0 
* 
* Unless required by applicable law or agreed to in writing, software 
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT 
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the 
* License for the specific language governing permissions and limitations 
* under the License. 
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/SslContextOption.java 
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License, 
# version 2.0 (the "License"); you may not use this file except in compliance 
# with the License. You may obtain a copy of the License at: 
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT 

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

// Try the OpenJDK's proprietary implementation.

// Found in path(s):

No license file was found, but licenses were detected in source scan.

*/

Copyright 2020 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

// Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/util/FingerprintTrustManagerFactoryBuilder.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/logging/ByteBufFormat.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources-jar/io/netty/handler/ssl/SslClosedEngineException.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources.jar/io/netty/handler/address/ResolveAddressHandler.java
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources.jar/io/netty/handler/ssl/SslHandshakeTimeoutException.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources.jar/io/netty/handler/traffic/GlobalChannelTrafficShapingHandler.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449572_1616156172.9/0/netty-handler-4-1-59-final-sources.jar/io/netty/handler/ssl/Util/X509TrustManagerWrapper.java
1.2219 rimraf 3.0.2

1.2219.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2220 base 0.11.2 0.11.2
1.2220.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2221 is-glob 2.0.1
1.2221.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2222 bitsyntax 0.1.0 0.1.0
1.2222.1 Available under license :

The MIT License (MIT)

Copyright (c) 2012-2014 Michael Bridgen <mikeb@squaremobius.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

bitsyntax-js copyright (c) 2012-2014 Michael Bridgen <mikeb@squaremobius.net>

This package, "bitsyntax-js", is licensed under the MIT License. A copy may be found in the file LICENSE-MIT in this directory, or downloaded from http://opensource.org/licenses/MIT.

1.2223 ansi-regex 5.0.0
1.2223.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2224 uzaygezen-core 0.2

1.2224.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/BitSetMath.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/BitSetBackedBitVector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/FilteredIndexRange.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/IndexCalculator.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/SpaceVisitor.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/MapRegionInspector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/SpaceVisitor.java
jar/com/google/uzaygezen/core/Query.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/SimpleRegionInspector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/MultiDimensionalSpec.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/RegionInspector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/LongBitVector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/PlainFilterCombiner.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/QueryFactory.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/CompactHilbertCurve.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/HilbertIndexMasks.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/TestUtils.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/ZoomingNavigator.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/LinkedList.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/BoundedRollup.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/SelectiveFilter.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/FilterCombiner.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/ZoomingSpaceVisitorAdapter.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/LongArrayBitVector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/BitVectorMath.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/ranges/LongRange.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/RangeListFilter.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/NodeList.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/SpatialRelation.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/Pow2LengthBitSetRange.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-jar/com/google/uzaygezen/core/BitVector.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/QueryBuilder.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/ListConcatCombiner.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/Pow2LengthBitSetRangeFactory.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/BitVectorFactories.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/MathUtils.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/Assessment.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/BitSetComparator.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/SpaceFillingCurve.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/AdditiveValue.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/BacktrackingQueryBuilder.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/MapNode.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/StreamingRollup.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/package-info.java
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/NodeValue.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 Daniel Aioanei.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446435_1631568163.64/0/uzaygezen-core-0-2-sources-
jar/com/google/uzaygezen/core/BigIntegerMath.java
1.2225 babel-plugin-transform-es2015-arrow-functions 6.22.0 6.22.0

1.2225.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2226 @babel/plugin-proposal-json-strings

7.16.0

1.2226.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2227 anymatch 3.1.1

1.2227.1 Available under license:
The ISC License

Copyright (c) 2019 Elan Shanker, Paul Miller (https://paulmillr.com)

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2228 commons-lang3 3.9

1.2228.1 Available under license:

Apache Commons Lang
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2229 compressible 2.0.14 2.0.14

1.2229.1 Available under license:
(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Jeremiah Senkpiel <fishrock123@rocketmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
1.2230 js-yaml 3.14.0
1.2230.1 Available under license:
(The MIT License)

Copyright (C) 2011-2015 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2231 type 2.1.0
1.2231.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Alejandro Celaya

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2232 p-locate 2.0.0
1.2232.1 Available under license :
   The MIT License (MIT)

   Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2233 minio 4.0.0

1.2234 fast-levenshtein 2.0.6 2.0.6
1.2234.1 Available under license :
   (MIT License)

   Copyright (c) 2013 [Ramesh Nair](http://www.hiddentao.com/)

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.2235 range-parser 1.2.1 1.2.1
1.2235.1 Available under license:
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2236 default-require-extensions 3.0.0

1.2236.1 Available under license:
MIT License

Copyright (c) Node.js contributors, James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2237 tephra-api 0.6.0

1.2237.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source
code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.2238.1 Available under license:
(The MIT License)

Copyright (c) 2012-2014 Roman Shtylman <shtylman@gmail.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2239 doctrine 2.1.0

1.2239.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2240.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling.
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:
a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.2241 repeat-element 1.1.3 1.1.3
1.2241.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2242 jersey-container-servlet-core 2.5.1

1.2242.1 Available under license:

CDDL+GPL License

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and commercial
computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

http://glassfish.java.net/public/CDDL+GPL_1_1.html
cat jersey-container-servlet-core-2.5.1-sources/org/glassfish/jersey/servlet/internal/ThreadLocalInvoker.java /*
* DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
*
* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved.
*
* The contents of this file are subject to the terms of either the GNU
* General Public License Version 2 only ("GPL") or the Common Development
* and Distribution License("CDDL") (collectively, the "License"). You
* may not use this file except in compliance with the License. You can
* obtain a copy of the License at
* http://glassfish.java.net/public/CDDL+GPL_1_1.html
* or packager/legal/LICENSE.txt. See the License for the specific
* language governing permissions and limitations under the License.
*
* When distributing the software, include this License Header Notice in each
* file and include the License file at packager/legal/LICENSE.txt.
*
* GPL Classpath Exception:
* Oracle designates this particular file as subject to the "Classpath"
* exception as provided by Oracle in the GPL Version 2 section of the License
* file that accompanied this code.
*
* Modifications:
* If applicable, add the following below the License Header, with the fields
* enclosed by brackets [] replaced by your own identifying information:
* "Portions Copyright [year] [name of copyright owner]"
*
* Contributor(s):
* If you wish your version of this file to be governed by only the CDDL or
* only the GPL Version 2, indicate your decision by adding "[Contributor]
* elects to include this software in this distribution under the [CDDL or GPL
* Version 2] license." If you don't indicate a single choice of license, a
* recipient has the option to distribute your version of this file under
* either the CDDL, the GPL Version 2 or to extend the choice of license to
* its licensees as provided above. However, if you add GPL Version 2 code
* and therefore, elected the GPL Version 2 license, then the option applies
* only if the new code is made subject to such option by the copyright
* holder.
*/
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU
General Public License is intended to guarantee your freedom to share and change free software--to make sure the
software is free for all its users. This General Public License applies to most of the Free Software Foundation's
software and to any other program whose authors commit to using it. (Some other Free Software Foundation
software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to
make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish),
that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library.
Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2243 trim-right 1.0.1 1.0.1

1.2243.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2244 pango 1.38.1-1

1.2244.1 Available under license:
GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and
distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these
For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying

Open Source Used In DNAC Platform Arica 10610
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
This package was debianized by Akira TAGOH <tagoh@debian.org> on Sun, 10 Mar 2002 22:30:33 +0900.

It was downloaded from:
Upstream Authors:
Owen Taylor <otaylor@redhat.com>

Robert Brady (Indic shapers), Hans Breuer (Windows backend), Matthias
Clasen (Documentation), Sivaraj Dossanan (Tamil shaper), Dov Grobgeld
(Hebrew shaper), Karl Koehler (Arabic shaper), Alex Larsson (FreeType
and Windows backends), Tor Lillqvist (FreeType and Windows backends),
Changwoo Ryu (Hangul shaper), Havoc Pennington, Roozbeh Pournader
(Arabic shaper), Chookij Vanatham (Hebrew shaper).

Copyright:
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this package; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

See /usr/share/common-licenses/LGPL-2 on your debian system.

The directory pango/mini-fribidi is copyright 1999,2000 Dov Grobgeld,
2001,2002 Behdad Esfahbod and is distributed under the following terms:

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License
along with this library, in a file named COPYING; if not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330,
Boston, MA 02111-1307, USA

On Debian systems the full text of the GNU Lesser General Public License can
be found in the `/usr/share/common-licenses/LGPL-2.1` file.

The file `tests/GraphemeBreakTest.txt` is copyright 1991-2014 Unicode, Inc and is distributed under the following terms:

Copyright 1991-2014 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

1.2245 jersey 2.25.1
1.2245.1 Available under license:

/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 * Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.
 * The contents of this file are subject to the terms of either the GNU
 * General Public License Version 2 only ("GPL") or the Common Development
* and Distribution License("CDDL") (collectively, the "License"). You
* may not use this file except in compliance with the License. You can
* obtain a copy of the License at
* http://glassfish.java.net/public/CDDL+GPL_1_1.html
* or packager/legal/LICENSE.txt. See the License for the specific
* language governing permissions and limitations under the License.
*
* When distributing the software, include this License Header Notice in each
* file and include the License file at packager/legal/LICENSE.txt.
*
* GPL Classpath Exception:
* Oracle designates this particular file as subject to the "Classpath"
* exception as provided by Oracle in the GPL Version 2 section of the License
* file that accompanied this code.
*
* Modifications:
* If applicable, add the following below the License Header, with the fields
* enclosed by brackets [ ] replaced by your own identifying information:
* "Portions Copyright [year] [name of copyright owner]"
*
* Contributor(s):
* If you wish your version of this file to be governed by only the CDDL or
* only the GPL Version 2, indicate your decision by adding "[Contributor]
* elects to include this software in this distribution under the [CDDL or GPL
* Version 2] license." If you don't indicate a single choice of license, a
* recipient has the option to distribute your version of this file under
* either the CDDL, the GPL Version 2 or to extend the choice of license to
* its licensees as provided above. However, if you add GPL Version 2 code
* and therefore, elected the GPL Version 2 license, then the option applies
* only if the new code is made subject to such option by the copyright
* holder.
*/

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.
1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the
terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.
7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or
modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and
    you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at
    compilers) written by James Hacker.

    signature of Ty Coon, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

DO NOT TRANSLATE OR LOCALIZE.

%% The following software may be included in this product: ASM
Use of any of this software is governed by the terms of the license below:
Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

%%% The following software may be included in this product: Jettison
Use of any of this software is governed by the terms of the license below:

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

%%% The following software may be included in this product: Jackson
Use of any of this software is governed by the terms of the license below:
Jackson is dual-licensed under two alternative popular Open Source licenses: Apache (AL 2.0) and Gnu Lesser GPL (LGPL 2.1). You choose one or the other, as necessary (if you want to redistribute the code for use, you do not need license), and abide by the license rules as defined by the respective license agreement (and only that one).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic
mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner] Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2246 http-status-codes 1.4.0 1.4.0

1.2246.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2247 cglib 2.2.0-b21

1.2247.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2004 The Apache Software Foundation
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
No license file was found, but licenses were detected in source scan.

2012 Oracle and/or its affiliates. All rights reserved.

The contents of this file are subject to the terms of either the GNU General Public License Version 2 only (“GPL”) or the Common Development and Distribution License (“CDDL”) (collectively, the "License"). You may not use this file except in compliance with the License. You can obtain a copy of the License at https://glassfish.dev.java.net/public/CDDL+GPL_1_1.html or packager/legal/LICENSE.txt. See the License for the specific language governing permissions and limitations under the License.

When distributing the software, include this License Header Notice in each file and include the License file at packager/legal/LICENSE.txt.

GPL Classpath Exception:
Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License file that accompanied this code.

Modifications:
If applicable, add the following below the License Header, with the fields enclosed by brackets [] replaced by your own identifying information:
"Portions Copyright [year] [name of copyright owner]"

Contributor(s):
If you wish your version of this file to be governed by only the CDDL or only the GPL Version 2, indicate your decision by adding "[Contributor]"
elects to include this software in this distribution under the [CDDL or GPL
Version 2] license." If you don't indicate a single choice of license, a
recipient has the option to distribute your version of this file under
either the CDDL, the GPL Version 2 or to extend the choice of license to
its licensees as provided above. However, if you add GPL Version 2 code
and therefore, elected the GPL Version 2 license, then the option applies
only if the new code is made subject to such option by the copyright
holder.

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/pom.xml
No license file was found, but licenses were detected in source scan.

/** *
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2005 INRIA, France Telecom
* All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/LocalVariablesSorter.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2003,2004 The Apache Software Foundation
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/DebuggingClassWriter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/LazyLoaderGenerator.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/CollectionUtils.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/CallbackGenerator.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/transform/ClassFilterTransformer.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/ReflectUtils.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/transform/impl/AddStaticInitTransformer.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/CodeEmitter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/MixinEmitter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/beans/BulkBeanEmitter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/MethodInterceptorGenerator.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/beans/ImmutableBean.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/beans/BeanCopier.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/DefaultNamingPolicy.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/CodeEmitter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/MixinEmitter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/core/EmitUtils.java
*/

* Copyright 2003,2004 The Apache Software Foundation
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
  * the License.
  */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/AbstractTransformTask.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2003 The Apache Software Foundation
*
* Licensed under the Apache License, Version 2.0 (the "License);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/AbstractClassTransformer.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/util/ParallelSorter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/MethodFilter.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/core/DefaultGeneratorStrategy.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/core/GeneratorStrategy.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/FieldVisitorTee.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/impl/UndeclaredThrowableTransformer.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/impl/UndeclaredThrowableStrategy.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/reflect/FastConstructor.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/reflect/ConstructorDelegate.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-
  jar/net/sf/cglib/transform/impl/AddPropertyTransformer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002,2003 The Apache Software Foundation
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/UndeclaredThrowableException.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/MethodInterceptor.java
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/Factory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002,2003,2004 The Apache Software Foundation
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136701410_1613864270.85/0/cglib-2-2-0-b21-sources-2-jar/net/sf/cglib/proxy/Enhancer.java
1.2248 snmp-4j 2.6.3

1.2248.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - JavaLogAdapter.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/log/JavaLogAdapter.java

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TLSTM.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/TLSTM.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/WorkerPool.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/MessageProcessingModel.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/MessageProcessingModel.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - SecureTarget.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/SecureTarget.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - AssignableFromIntArray.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/smi/AssignableFromIntArray.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - StatusInformation.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/mp/StatusInformation.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - DHOperations.java
_##  Copyright (C) 2003-2017 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_########################################################################*/
## See the License for the specific language governing permissions and limitations under the License.

### SNMP4J 2 - DummyTransport.java

### Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

### Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

### You may obtain a copy of the License at

### http://www.apache.org/licenses/LICENSE-2.0

### Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

### See the License for the specific language governing permissions and limitations under the License.

### SNMP4J 2 - TsmSecurityStateReference.java

### Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

### Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

### You may obtain a copy of the License at

### http://www.apache.org/licenses/LICENSE-2.0

### Unless required by applicable law or agreed to in writing, software
Open Source Used In DNAC Platform Arica 10650

_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/uri/SnmpUriCallback.java
No license file was found, but licenses were detected in source scan.

/"_##########################################################################
_##
_## SNMP4J 2 - AuthHMAC128SHA224.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/AuthHMAC128SHA224.java
No license file was found, but licenses were detected in source scan.

/"_##########################################################################
_##
_## SNMP4J 2 - Snmp.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
Open Source Used In DNAC Platform Arica 10651

_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/Snmp.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TransportStateReference.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/TransportStateReference.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - CipherPool.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/SNMP4JSettings.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - AuthenticationProtocol.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/AuthenticationProtocol.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - RetrievalEvent.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##
_########################################################################*/
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/RetrievalEvent.java
No license file was found, but licenses were detected in source scan.

SNMP4J 2 - PrivacyProtocol.java
Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/PrivacyProtocol.java
No license file was found, but licenses were detected in source scan.

SNMP4J 2 - DefaultUdpTransportMapping.java
Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

########################################################################*/

Found in path(s):
* /opt/ cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/transport/DefaultUdpTransportMapping.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - DefaultTlsTmSecurityCallback.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
_##
_############################################################################*/

Found in path(s):
* /opt/ cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/transport/tls/DefaultTlsTmSecurityCallback.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - ResponseEvent.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
Open Source Used In DNAC Platform Arica 10657

__## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)  
__##  
__## Licensed under the Apache License, Version 2.0 (the "License");  
__## you may not use this file except in compliance with the License.  
__## You may obtain a copy of the License at  
__##  
__## http://www.apache.org/licenses/LICENSE-2.0  
__##  
__## Unless required by applicable law or agreed to in writing, software  
__## distributed under the License is distributed on an "AS IS" BASIS,  
__## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
__## See the License for the specific language governing permissions and  
__## limitations under the License.  
__##  
__## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/Salt.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - TransportMapping.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/TransportMapping.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - SimpleVariableTextFormat.java
_##
/* _############################################################################
_##
_##  SNMP4J 2 - JavaLogFactory.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/log/JavaLogFactory.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_##
_##  SNMP4J 2 - SshSession.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/ssh/SshSession.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
###
### SNMP4J 2 - AbstractTarget.java
###
### Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
###
### Licensed under the Apache License, Version 2.0 (the "License");
### you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
###
### http://www.apache.org/licenses/LICENSE-2.0
###
### Unless required by applicable law or agreed to in writing, software
### distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.
###
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/AbstractTarget.java

No license file was found, but licenses were detected in source scan.

/* _############################################################################
###
### SNMP4J 2 - MaxAccess.java
###
### Copyright (C) 2003-2016 Frank Fock (SNMP4J.org)
###
### Licensed under the Apache License, Version 2.0 (the "License");
### you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
###
### http://www.apache.org/licenses/LICENSE-2.0
###
### Unless required by applicable law or agreed to in writing, software
### distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.
###
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/MaxAccess.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_## SNMP4J 2 - MessageID.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/MessageID.java

No license file was found, but licenses were detected in source scan.

/* _############################################################################
_## SNMP4J 2 - Address.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Address.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_##
_##  SNMP4J 2 - SshAddress.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/SshAddress.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_##
_##  SNMP4J 2 - SecurityLevel.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/SecurityLevel.java
jar/org/snmp4j/security/SecurityLevel.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_## SNMP4J 2 - LogLevel.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_## _############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/log/LogLevel.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_## SNMP4J 2 - RequestStatistics.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_## _############################################################################*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - BEROutputStream.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_########################################################################*/

Found in path(s):

/*_############################################################################
_## SNMP4J 2 - CommandResponder.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/CommandResponder.java
No license file was found, but licenses were detected in source scan.

/"_############################################################################
_##
_## SNMP4J 2 - VersionInfo.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/version/VersionInfo.java
No license file was found, but licenses were detected in source scan.

/"_############################################################################
_##
_## SNMP4J 2 - CounterEvent.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/
Open Source Used In DNAC Platform Arica 10669

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/event/CounterEvent.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - SubIndexInfo.java
_## Copyright (C) 2003-2016 Frank Fock (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/SubIndexInfo.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - VariantVariable.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/TimeTicks.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_##
_##  SNMP4J 2 - SMIConstants.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/SMIConstants.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_##
_##  SNMP4J 2 - SnmpConstants.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
########################################################################*/
/*_############################################################################
_## SNMP4J 2 - BERInputStream.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/SnmpConstants.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - OID.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/asn1/BERInputStream.java
No license file was found, but licenses were detected in source scan.
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/OID.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/Priv3DES.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/TcpTransportMapping.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmpl/transport/TcpTransportMapping.java
No license file was found, but licenses were detected in source scan.

/* _##########################################################################*/
## SNMP4J 2 - TransportIpAddress.java
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
## _##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmpl/smi/TransportIpAddress.java
No license file was found, but licenses were detected in source scan.

/* _##########################################################################*/
## SNMP4J 2 - ConsoleLogFactory.java
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
Open Source Used In DNAC Platform Arica 10675

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/log/ConsoleLogFactory.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/log/ConsoleLogAdapter.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/log/SnmpRequest.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SNMP4J 2 - TableEvent.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SNMP4J 2 - PrivDES.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/PrivDES.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_## SNMP4J 2 - MessageDispatcherImpl.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_##   http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/MessageDispatcherImpl.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_## SNMP4J 2 - SchedulerTask.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_##   http://www.apache.org/licenses/LICENSE-2.0
_##
_##########################################################################*/
## License

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/SchedulerTask.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

## Copyright

SNMP4J 2 - TlsTmSecurityCallback.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/tls/TlsTmSecurityCallback.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

## Copyright

SNMP4J 2 - SimpleMessageID.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Open Source Used In DNAC Platform Arica 10680

---

No license file was found, but licenses were detected in source scan.

---

No license file was found, but licenses were detected in source scan.

---

No license file was found, but licenses were detected in source scan.
Open Source Used In DNAC Platform Arica 10681

---

```
##
##  http://www.apache.org/licenses/LICENSE-2.0
##
##  Unless required by applicable law or agreed to in writing, software
##  distributed under the License is distributed on an "AS IS" BASIS,
##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##  See the License for the specific language governing permissions and
##  limitations under the License.
##
## ##########################################################################

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/AuthHMAC256SHA384.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - Gauge32.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Gauge32.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - PropertiesTlsTmSecurityCallback.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##
_##########################################################################

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/PropertiesTlsTmSecurityCallback.java
No license file was found, but licenses were detected in source scan.

/*
_##########################################################################
_##
_##  SNMP4J 2 - PropertiesTlsTmSecurityCallback.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##
_##########################################################################
```
Open Source Used In DNAC Platform Arica 10683

### Licensed under the Apache License, Version 2.0 (the "License");
### you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
###
### http://www.apache.org/licenses/LICENSE-2.0
###
### Unless required by applicable law or agreed to in writing, software
### distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.
###
#############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/TableUtils.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TransportStateEvent.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/TransportStateEvent.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - ThreadPool.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/ThreadPool.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - ThreadPool.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
/* _############################################################################
_## SNMP4J 2 - BitString.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_## _##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/BitString.java
No license file was found, but licenses were detected in source scan.

/* _############################################################################
_## SNMP4J 2 - CommonTimer.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_## _##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/CommonTimer.java
No license file was found, but licenses were detected in source scan.
/*##############################################################################
  ## SNMP4J 2 - VariantVariableCallback.java
  ##
  ## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
  ##
  ## Licensed under the Apache License, Version 2.0 (the "License");
  ## you may not use this file except in compliance with the License.
  ## You may obtain a copy of the License at
  ##
  ## http://www.apache.org/licenses/LICENSE-2.0
  ##
  ## Unless required by applicable law or agreed to in writing, software
  ## distributed under the License is distributed on an "AS IS" BASIS,
  ## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  ## See the License for the specific language governing permissions and
  ## limitations under the License.
  ##
  ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/smi/VariantVariableCallback.java

No license file was found, but licenses were detected in source scan.

/*##############################################################################
  ## SNMP4J 2 - SecurityModels.java
  ##
  ## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
  ##
  ## Licensed under the Apache License, Version 2.0 (the "License");
  ## you may not use this file except in compliance with the License.
  ## You may obtain a copy of the License at
  ##
  ## http://www.apache.org/licenses/LICENSE-2.0
  ##
  ## Unless required by applicable law or agreed to in writing, software
  ## distributed under the License is distributed on an "AS IS" BASIS,
  ## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  ## See the License for the specific language governing permissions and
  ## limitations under the License.
  ##
  ##########################################################################*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * _############################################################################
 * _##  SNMP4J 2 - SnmpConfigurator.java
 * _##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
 * _##  Licensed under the Apache License, Version 2.0 (the "License");
 * _##  you may not use this file except in compliance with the License.
 * _##  You may obtain a copy of the License at
 * _##
 * _##      http://www.apache.org/licenses/LICENSE-2.0
 * _##
 * _##  Unless required by applicable law or agreed to in writing, software
 * _##  distributed under the License is distributed on an "AS IS" BASIS,
 * _##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * _##  See the License for the specific language governing permissions and
 * _##  limitations under the License.
 * _##
 * _##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/util/SnmpConfigurator.java
No license file was found, but licenses were detected in source scan.

/*
 * _##########################################################################*/
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - ThreadFactory.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_########################################################################*/

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - Opaque.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Opaque.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Opaque.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/test/MultiThreadedTrapReceiver.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
Open Source Used In DNAC Platform Arica 10695

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/nonstandard/PrivAES256With3DESExtension.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TimeoutModel.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/TimeoutModel.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TimedMessageID.java
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
SNMP4J 2 - DefaultTimeoutModel.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

SNMP4J 2 - BER.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.
###
### ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/asn1/BER.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - CommandResponderEvent.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##  ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/CommandResponderEvent.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - LogFactory.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

### Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Null.java

No license file was found, but licenses were detected in source scan.

/* ###################################################################
 **
 ** SNMP4J 2 - EnumerationIterator.java
 **
 ** Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
 **
 ** Licensed under the Apache License, Version 2.0 (the "License");
 ** you may not use this file except in compliance with the License.
 ** You may obtain a copy of the License at
 **
 ** http://www.apache.org/licenses/LICENSE-2.0
 **
 ** Unless required by applicable law or agreed to in writing, software
 ** distributed under the License is distributed on an "AS IS" BASIS,
 ** WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 ** See the License for the specific language governing permissions and
 ** limitations under the License.
 **
 ** ###################################################################*/

### Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/EnumerationIterator.java

No license file was found, but licenses were detected in source scan.

/* ###################################################################
 **
 ** SNMP4J 2 - DefaultCounterListener.java
 **
 ** Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
 **
 ** Licensed under the Apache License, Version 2.0 (the "License");
 ** you may not use this file except in compliance with the License.
 ** You may obtain a copy of the License at
 **
 ** http://www.apache.org/licenses/LICENSE-2.0
 **
 ** ###################################################################*/
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/DefaultCounterListener.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
_## SNMP4J 2 - NoLogger.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_## ##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/log/NoLogger.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
_## SNMP4J 2 - WorkerTask.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0

_##
_##
_##
_##
_##
_##
_##
_##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/WorkerTask.java
No license file was found, but licenses were detected in source scan.

/"_############################################################################
_##
_## SNMP4J 2 - TreeUtils.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/TreeUtils.java
No license file was found, but licenses were detected in source scan.

2004 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE
2.0
_##
_## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
Open Source Used In DNAC Platform Arica 10702

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/asn1/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/event/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/package.html

No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/SecurityParameters.java

No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/SecurityParameters.java

No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/SecurityParameters.java

No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/SecurityParameters.java

No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/TreeListener.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - PrivAES192.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/PrivAES192.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - DefaultTcpTransportMapping.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
_##
_########################################################################*/
You may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/security/SecurityStateReference.java
No license file was found, but licenses were detected in source scan.

/"_############################################################################
_##  SNMP4J 2 - SnmpEngineListener.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
_##
_############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/event/SnmpEngineListener.java
No license file was found, but licenses were detected in source scan.

/"_############################################################################
_##  SNMP4J 2 - Variable.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0

## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/smi/AssignableFromByteArray.java

No license file was found, but licenses were detected in source scan.

/*/ __________________________________________________________

## SNMP4J 2 - VariableBinding.java

## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0

## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/VariableBinding.java

No license file was found, but licenses were detected in source scan.

/*/ __________________________________________________________

## SNMP4J 2 - UsmTimeTable.java
### SNMP4J 2 - LogAdapter.java

### Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

### Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
### http://www.apache.org/licenses/LICENSE-2.0

### Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and limitations under the License.

##########################################################################

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/log/LogAdapter.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - CommunityTarget.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/CommunityTarget.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/AuthSHA.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - MPv2c.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##  ############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/MPv2c.java
No license file was found, but licenses were detected in source scan.
Open Source Used In DNAC Platform Arica 10713
No license file was found, but licenses were detected in source scan.

/* _############################################################################
##
##  SNMP4J 2 - OIDTextFormat.java
##
##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
##
##  Licensed under the Apache License, Version 2.0 (the "License");
##  you may not use this file except in compliance with the License.
##  You may obtain a copy of the License at
##
##  http://www.apache.org/licenses/LICENSE-2.0
##
##  Unless required by applicable law or agreed to in writing, software
##  distributed under the License is distributed on an "AS IS" BASIS,
##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##  See the License for the specific language governing permissions and
##  limitations under the License.
##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/OIDTextFormat.java

No license file was found, but licenses were detected in source scan.

/* _############################################################################
##
##  SNMP4J 2 - PDU.java
##
##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
##
##  Licensed under the Apache License, Version 2.0 (the "License");
##  you may not use this file except in compliance with the License.
##  You may obtain a copy of the License at
##
##  http://www.apache.org/licenses/LICENSE-2.0
##
##  Unless required by applicable law or agreed to in writing, software
##  distributed under the License is distributed on an "AS IS" BASIS,
##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##  See the License for the specific language governing permissions and
##  limitations under the License.
##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/PDU.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - DefaultPDUFactory.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/util/DefaultPDUFactory.java

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - TimerFactory.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*_############################################################################ 
_##  SNMP4J 2 - AuthMD5.java 
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org) 
_##  Licensed under the Apache License, Version 2.0 (the "License"); 
_##  you may not use this file except in compliance with the License. 
_##  You may obtain a copy of the License at 
_##  http://www.apache.org/licenses/LICENSE-2.0 
_##  Unless required by applicable law or agreed to in writing, software 
_##  distributed under the License is distributed on an "AS IS" BASIS, 
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
_##  See the License for the specific language governing permissions and 
_##  limitations under the License. 
_## 
############################################################################*/

Found in path(s):

/*_############################################################################ 
_##  SNMP4J 2 - TableListener.java 
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org) 
_##  Licensed under the Apache License, Version 2.0 (the "License"); 
_##  you may not use this file except in compliance with the License. 
_##  You may obtain a copy of the License at 
_##  http://www.apache.org/licenses/LICENSE-2.0 
_##  Unless required by applicable law or agreed to in writing, software 
_##  distributed under the License is distributed on an "AS IS" BASIS, 
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
_##  See the License for the specific language governing permissions and 
_##  limitations under the License. 
_## 
############################################################################*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - AssignableFromInteger.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1.jar/org/snmp4j/smi/AssignableFromInteger.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_## SNMP4J 2 - Log4jLogFactory.java
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_## http://www.apache.org/licenses/LICENSE-2.0
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/log/Log4jLogFactory.java

No license file was found, but licenses were detected in source scan.

/* ############################################################################
   ## SNMP4J 2 - SecurityProtocols.java
   ## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
   ## Licensed under the Apache License, Version 2.0 (the "License");
   ## you may not use this file except in compliance with the License.
   ## You may obtain a copy of the License at
   ##     http://www.apache.org/licenses/LICENSE-2.0
   ## Unless required by applicable law or agreed to in writing, software
   ## distributed under the License is distributed on an "AS IS" BASIS,
   ## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   ## See the License for the specific language governing permissions and
   ## limitations under the License.
   ## ############################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/SecurityProtocols.java

No license file was found, but licenses were detected in source scan.

/* ############################################################################
   ## SNMP4J 2 - AuthenticationFailureEvent.java
   ## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
   ## Licensed under the Apache License, Version 2.0 (the "License");
   ## you may not use this file except in compliance with the License.
   ## You may obtain a copy of the License at
   ##     http://www.apache.org/licenses/LICENSE-2.0
   ## Unless required by applicable law or agreed to in writing, software
   ## distributed under the License is distributed on an "AS IS" BASIS,
   ## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   ## See the License for the specific language governing permissions and
   ## limitations under the License.
   ## ############################################################################*/
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - SshTransportAdapter.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_########################################################################*/

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - DefaultSshTransportMapping.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##      http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
Open Source Used in DNAC Platform Arica 10720

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/DefaultSshTransportMapping.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - Session.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/Session.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################
_##
_##  SNMP4J 2 - TlsAddress.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.40/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/TlsAddress.java
No license file was found, but licenses were detected in source scan.

/*_################################################################################
__
___SNMP4J 2 - TreeEvent.java
___
____Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
____
____Licensed under the Apache License, Version 2.0 (the "License");
____you may not use this file except in compliance with the License.
____You may obtain a copy of the License at
____
____http://www.apache.org/licenses/LICENSE-2.0
____
____Unless required by applicable law or agreed to in writing, software
____distributed under the License is distributed on an "AS IS" BASIS,
____WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
____See the License for the specific language governing permissions and
____limitations under the License.
____
_################################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.40/snmp4j-2-6-3-sources-1-jar/org/snmp4j/util/TreeEvent.java
No license file was found, but licenses were detected in source scan.

/*_################################################################################
__
___SNMP4J 2 - TSM.java
___
____Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
____
____Licensed under the Apache License, Version 2.0 (the "License");
____you may not use this file except in compliance with the License.
____You may obtain a copy of the License at
____
____http://www.apache.org/licenses/LICENSE-2.0
____
____Unless required by applicable law or agreed to in writing, software
____distributed under the License is distributed on an "AS IS" BASIS,
____WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
____See the License for the specific language governing permissions and
No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.

No license file was found, but licenses were detected in source scan.
## Open Source Used In DNAC Platform

<table>
<thead>
<tr>
<th>License Type</th>
<th>License Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache 2.0</td>
<td>See the License for the specific language governing permissions and limitations under the License.</td>
</tr>
</tbody>
</table>

**Found in path(s):**

* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/nonstandard/PrivAES192With3DESKeyExtension.java

No license file was found, but licenses were detected in source scan.

```java
/**
 * SNMP4J 2 - PrivAES256.java
 */

## SNMP4J 2 - PrivAES256.java

## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0

## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.

**Found in path(s):**

* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/PrivAES256.java

No license file was found, but licenses were detected in source scan.

```java
/**
 * SNMP4J 2 - MultiThreadedMessageDispatcher.java
 */

## SNMP4J 2 - MultiThreadedMessageDispatcher.java

## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0

## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SNMP4J 2 - TransportMappings.java

Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SNMP4J 2 - ResponseListener.java

Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0
## Open Source Used In DNAC Platform Arica 10726

```
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/event/ResponseListener.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
##
## SNMP4J 2 - AssignableFromString.java
##
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/smi/AssignableFromString.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
##
## SNMP4J 2 - AuthGeneric.java
##
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
##########################################################################*/
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/AuthGeneric.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - SNMPv3SecurityModel.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/security/TLSTMExtendedTrustManagerFactory.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - TLSTMExtendedTrustManagerFactory.java
_##
_##  Copyright (C) 2003-2017  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SNMP4J 2 - PduHandle.java
Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SNMP4J 2 - CertifiedTarget.java
Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at 
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and 
limitations under the License.

########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/CertifiedTarget.java
No license file was found, but licenses were detected in source scan.

/**_############################################################################
_##  SNMP4J 2 - StateReference.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and 
limitations under the License.
_##
########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/mp/StateReference.java
No license file was found, but licenses were detected in source scan.

/**_############################################################################
_##  SNMP4J 2 - DHPARAMETERS.java
_##
_##  Copyright (C) 2003-2017 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
## SNMP4J 2 - MPv3.java

### Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)

### Licensed under the Apache License, Version 2.0 (the "License");
### you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
###
###   http://www.apache.org/licenses/LICENSE-2.0

### Unless required by applicable law or agreed to in writing, software
### distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.

########################################################################*/

No license file was found, but licenses were detected in source scan.

/*_########################################################################*/

## SNMP4J 2 - UnsignedInteger32.java

### Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)

### Licensed under the Apache License, Version 2.0 (the "License");
### you may not use this file except in compliance with the License.
### You may obtain a copy of the License at
###
###   http://www.apache.org/licenses/LICENSE-2.0

### Unless required by applicable law or agreed to in writing, software
### distributed under the License is distributed on an "AS IS" BASIS,
### WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
### See the License for the specific language governing permissions and
### limitations under the License.

########################################################################*/

No license file was found, but licenses were detected in source scan.
/*_############################################################################
_## SNMP4J 2 - AssignableFromLong.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_##   http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/AssignableFromLong.java
No license file was found, but licenses were detected in source scan.

2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_##   http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/package.html
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/transport/package.html
No license file was found, but licenses were detected in source scan.

/*_############################################################################
SNMP4J 2 - SnmpURI.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SNMP4J 2 - MPv1.java

Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
/*_############################################################################
_##  SNMP4J 2 - OctetString.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/OctetString.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - CertifiedIdentity.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/CertifiedIdentity.java
No license file was found, but licenses were detected in source scan.
/*_##########################################################################
_**##  SNMP4J 2 - UsmUserListener.java
_**##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_**##
_**##  Licensed under the Apache License, Version 2.0 (the "License");
_**##  you may not use this file except in compliance with the License.
_**##  You may obtain a copy of the License at
_**##
_**##  http://www.apache.org/licenses/LICENSE-2.0
_**##
_**##  Unless required by applicable law or agreed to in writing, software
_**##  distributed under the License is distributed on an "AS IS" BASIS,
_**##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_**##  See the License for the specific language governing permissions and
_**##  limitations under the License.
_**##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/event/UsmUserListener.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################
_**##  SNMP4J 2 - IpAddress.java
_**##
_**##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_**##
_**##  Licensed under the Apache License, Version 2.0 (the "License");
_**##  you may not use this file except in compliance with the License.
_**##  You may obtain a copy of the License at
_**##
_**##  http://www.apache.org/licenses/LICENSE-2.0
_**##
_**##  Unless required by applicable law or agreed to in writing, software
_**##  distributed under the License is distributed on an "AS IS" BASIS,
_**##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_**##  See the License for the specific language governing permissions and
_**##  limitations under the License.
_**##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/IpAddress.java

Open Source Used In DNAC Platform Arica 10738
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - AbstractVariable.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmtp4j/smi/AbstractVariable.java

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_##  SNMP4J 2 - UsmUserEntry.java
_##
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##    http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - Counter64.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/smi/Counter64.java

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - UsmUser.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##  http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
*/
__############################################################################
##
##  SNMP4J 2 - UsmSecurityParameters.java
##
##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
##  Licensed under the Apache License, Version 2.0 (the "License");
##  you may not use this file except in compliance with the License.
##  You may obtain a copy of the License at
##
##     http://www.apache.org/licenses/LICENSE-2.0
##
##  Unless required by applicable law or agreed to in writing, software
##  distributed under the License is distributed on an "AS IS" BASIS,
##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##  See the License for the specific language governing permissions and
##  limitations under the License.
##
__##########################################################################*/

Found in path(s):
*/
__############################################################################
##
##  SNMP4J 2 - UserTarget.java
##
##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
##  Licensed under the Apache License, Version 2.0 (the "License");
##  you may not use this file except in compliance with the License.
##  You may obtain a copy of the License at
##
##     http://www.apache.org/licenses/LICENSE-2.0
##
##  Unless required by applicable law or agreed to in writing, software
##  distributed under the License is distributed on an "AS IS" BASIS,
##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##  See the License for the specific language governing permissions and
##  limitations under the License.
##
__##########################################################################*/
/*  ###################################################################################################################
   ##  SNMP4J 2 - MutableStateReference.java
   ##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
   ##  Licensed under the Apache License, Version 2.0 (the "License");
   ##  you may not use this file except in compliance with the License.
   ##  You may obtain a copy of the License at
   ##      http://www.apache.org/licenses/LICENSE-2.0
   ##  Unless required by applicable law or agreed to in writing, software
   ##  distributed under the License is distributed on an "AS IS" BASIS,
   ##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   ##  See the License for the specific language governing permissions and
   ##  limitations under the License.
   ##  ###################################################################################################################*/

/*  ###################################################################################################################
   ##  SNMP4J 2 - MutablePDU.java
   ##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
   ##  Licensed under the Apache License, Version 2.0 (the "License");
   ##  you may not use this file except in compliance with the License.
   ##  You may obtain a copy of the License at
   ##      http://www.apache.org/licenses/LICENSE-2.0
   ##  Unless required by applicable law or agreed to in writing, software
   ##  distributed under the License is distributed on an "AS IS" BASIS,
   ##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   ##  See the License for the specific language governing permissions and
   ##  limitations under the License.
   ##  ###################################################################################################################*/
Open Source Used In DNAC Platform Arica 10743

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-jar/org/snmp4j/MutablePDU.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - SnmpEngineEvent.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##     http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/event/SnmpEngineEvent.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - UdpTransportMapping.java
_##  Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##     http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/transport/UdpTransportMapping.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
##
## SNMP4J 2 - PduHandleCallback.java
##
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
## limitations under the License.
##
##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/mp/PduHandleCallback.java
No license file was found, but licenses were detected in source scan.

/*_##########################################################################*/
##
## SNMP4J 2 - MessageLength.java
##
## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
##
## Licensed under the Apache License, Version 2.0 (the "License");
## you may not use this file except in compliance with the License.
## You may obtain a copy of the License at
##
## http://www.apache.org/licenses/LICENSE-2.0
##
## Unless required by applicable law or agreed to in writing, software
## distributed under the License is distributed on an "AS IS" BASIS,
## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
## See the License for the specific language governing permissions and
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - TransportListener.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##     http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
_##  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_##  See the License for the specific language governing permissions and
_##  limitations under the License.
_##
_########################################################################*/

No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - DecryptParams.java
_##
_##  Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_##  Licensed under the Apache License, Version 2.0 (the "License");
_##  you may not use this file except in compliance with the License.
_##  You may obtain a copy of the License at
_##
_##     http://www.apache.org/licenses/LICENSE-2.0
_##
_##  Unless required by applicable law or agreed to in writing, software
_##  distributed under the License is distributed on an "AS IS" BASIS,
Open Source Used In DNAC Platform Arica 10746

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/security/DecryptParams.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_## SNMP4J 2 - AuthenticationFailureListener.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_## Unless required by applicable law or agreed to in writing, software
_## distributed under the License is distributed on an "AS IS" BASIS,
_## WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
_## See the License for the specific language governing permissions and
_## limitations under the License.
_##
_##########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
jar/org/snmp4j/event/AuthenticationFailureListener.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
_## SNMP4J 2 - UnsupportedAddressClassException.java
_##
_## Copyright (C) 2003-2016 Frank Fock and Jochen Katz (SNMP4J.org)
_##
_## Licensed under the Apache License, Version 2.0 (the "License");
_## you may not use this file except in compliance with the License.
_## You may obtain a copy of the License at
_##
_## http://www.apache.org/licenses/LICENSE-2.0
_##
_##########################################################################*/
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/util/DefaultTimerFactory.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - PrivAES128.java
_##
##_ Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
##_ Licensed under the Apache License, Version 2.0 (the "License");
##_ you may not use this file except in compliance with the License.
##_ You may obtain a copy of the License at
_##
##_ http://www.apache.org/licenses/LICENSE-2.0
_##
##_ Unless required by applicable law or agreed to in writing, software
##_ distributed under the License is distributed on an "AS IS" BASIS,
##_ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
##_ See the License for the specific language governing permissions and
##_ limitations under the License.
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/security/PrivAES128.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##  SNMP4J 2 - MessageException.java
_##
##_ Copyright (C) 2003-2016  Frank Fock and Jochen Katz (SNMP4J.org)
_##
##_ Licensed under the Apache License, Version 2.0 (the "License");
##_ you may not use this file except in compliance with the License.
##_ You may obtain a copy of the License at
_##
_########################################################################*/

Found in path(s):
* /opt/cola/permits/1136700372_1613864639.4/0/snmp4j-2-6-3-sources-1-
  jar/org/snmp4j/security/PrivAES128.java
No license file was found, but licenses were detected in source scan.

/*_############################################################################
_##
##_ Open Source Used In DNAC Platform Arica 10748
_########################################################################*/
1. Definitions.

1.1. Contributor, means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You
distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of
the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as a Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING
NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

   The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

   If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions.
You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2250 **https-proxy-agent 2.2.4**

1.2250.1 Available under license:

No license file was found, but licenses were detected in source scan.

https-proxy-agent
### An HTTP(s) proxy `http.Agent` implementation for HTTPS

This module provides an `http.Agent` implementation that connects to a specified HTTP or HTTPS proxy server, and can be used with the built-in `https` module.

Specifically, this `Agent` implementation connects to an intermediary "proxy" server and issues the [CONNECT HTTP method]([CONNECT](https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/CONNECT)), which tells the proxy to open a direct TCP connection to the destination server.

Since this agent implements the CONNECT HTTP method, it also works with other protocols that use this method when connecting over proxies (i.e. WebSockets). See the "Examples" section below for more.

Installation
------------

Install with `npm`:

```bash
$ npm install https-proxy-agent
```

Examples
--------

#### `https` module example

```js
var url = require('url');
var https = require('https');
var HttpsProxyAgent = require('https-proxy-agent');

// HTTP/HTTPS proxy to connect to
var proxy = process.env.http_proxy || 'http://168.63.76.32:3128';
console.log('using proxy server %j', proxy);

// HTTPS endpoint for the proxy to connect to
console.log('attempting to GET %j', endpoint);
var options = url.parse(endpoint);

// create an instance of the `HttpsProxyAgent` class with the proxy server information
var agent = new HttpsProxyAgent(proxy);
```

---

Open Source Used In DNAC Platform Arica 10761
options.agent = agent;

https.get(options, function (res) {
    console.log("response" event!, res.headers);
    res.pipe(process.stdout);
});

``` js
var url = require('url');
var WebSocket = require('ws');
var HttpsProxyAgent = require('https-proxy-agent');

// HTTP/HTTPS proxy to connect to
var proxy = process.env.http_proxy || 'http://168.63.76.32:3128';
console.log('using proxy server %j', proxy);

// WebSocket endpoint for the proxy to connect to
var endpoint = process.argv[2] || 'ws://echo.websocket.org';
var parsed = url.parse(endpoint);
console.log('attempting to connect to WebSocket %j', endpoint);

// create an instance of the `HttpsProxyAgent` class with the proxy server information
var options = url.parse(proxy);
var agent = new HttpsProxyAgent(options);

// finally, initiate the WebSocket connection
var socket = new WebSocket(endpoint, { agent: agent });

socket.on('open', function () {
    console.log("open" event!);
    socket.send('hello world');
});

socket.on('message', function (data, flags) {
    console.log("message" event! %j %j', data, flags);
    socket.close();
});
```

API
---

``` js
### new HttpsProxyAgent(Object options)
```
The `HttpsProxyAgent` class implements an `http.Agent` subclass that connects to the specified "HTTP(s) proxy server" in order to proxy HTTPS and/or WebSocket requests. This is achieved by using the [HTTP ‘CONNECT’ method][CONNECT].

The `options` argument may either be a string URI of the proxy server to use, or an "options" object with more specific properties:

* `host` - String - Proxy host to connect to (may use `hostname` as well). Required.
* `port` - Number - Proxy port to connect to. Required.
* `protocol` - String - If `https:`, then use TLS to connect to the proxy.
* `headers` - Object - Additional HTTP headers to be sent on the HTTP CONNECT method.
* Any other options given are passed to the `net.connect()`/'tls.connect()` functions.

License
------

(The MIT License)

Copyright (c) 2013 Nathan Rajlich &lt;nathan@tootallnate.net&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[CONNECT]: http://en.wikipedia.org/wiki/HTTP_tunnel#HTTP_CONNECT_Tunneling

Found in path(s):
* /opt/cola/permits/1112087480_1607035434.28/0/https-proxy-agent-2-2-4-6-tgz/package/README.md
1.2251 commons-compress 2.1.1 2.1.1

1.2251.1 Available under license:

Copyright (c) 2014 Chris Talkington, contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2252 lcid 2.0.0 2.0.0

1.2252.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2253 libedit 3.1-20150325-1ubuntu2

1.2253.1 Available under license:

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

1.2254 global-dirs 3.0.0

1.2254.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2255 http-errors 1.6.3 1.6.3
1.2255.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com
Copyright (c) 2016 Douglas Christopher Wilson doug@somethingdoug.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2256 orc-tools 1.5.6
1.2256.1 Available under license:

ORC Tools
Copyright 2013-2019 The Apache Software Foundation

This product includes software developed at
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Open Source Used In DNAC Platform Arica 10770
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2257 hive-llap-common 3.1.2
1.2257.1 Available under license :

Hive Llap Common
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2258 cookiejar 2.0.6
1.2258.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013 Bradley Meck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2259 array prototype flat 1.2.5

1.2259.1 Available under license :
MIT License

Copyright (c) 2017 ECMAScript Shims

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2260 @babel/preset-modules 0.1.5

1.2260.1 Available under license :
MIT License

Copyright (c) 2020 Babel

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2261 ratelimitj-inmemory 0.4.0

1.2261.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

 http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2262 jets3t 0.6.1

1.2262.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
 * This is a java.net project, see https://jets3t.dev.java.net/
 *
 * Copyright 2007 James Murty
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
No license file was found, but licenses were detected in source scan.

/*
 * jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
 * This is a java.net project, see https://jets3t.dev.java.net/
 * Copyright 2006 James Murty
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/org/jets3t/service/io/GZipInflatingOutputStream.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/org/jets3t/gui/skins/html/SkinnedJComboBox.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/org/jets3t/service/utils/gatekeeper/GatekeeperMessage.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/org/jets3t/service/utils/gatekeeper/SignatureRequest.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/io/ProgressMonitoredInputStream.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/impl/rest/httpclient/RestS3Service.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/model/S3Object.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/gui/skins/html/SkinnedJTextArea.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/acl/EmailAddressGrantee.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/impl/rest/RestS3Service.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/model/S3ObjectChunk.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/utils/RestUtils.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/sessions/OAuthApplication.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/sessions/OAuthApplicationImpl.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/sessions/OAuthApplicationImplService.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
  jar/src/jets3t/service/sessions/SessionForOAuthApplications.java
jar/src/org/jets3t/service/multithread/DownloadPackage.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/io/InputStreamWrapper.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/model/S3Bucket.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/io/InterruptableInputStream.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/acl/Permission.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/multithread/CancelEventTrigger.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/samples/CodeSamples.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/gui/GuiUtils.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/model/BaseS3Object.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/gui/skins/html/SkinnedJRadioButton.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/impl/rest/httpclient/RepeatableRequestEntity.java
* /opt/cola/permits/1139709450_1614684284.5/0/jets3t-0-6-1-sources-2-
jar/src/org/jets3t/service/utils/FileComparer.java

No license file was found, but licenses were detected in source scan.

/*
 * $HeadURL:
https://svn.apache.org/repos/asf/jakarta/commons/proper/httpclient/branches/HTTPCLIENT_3_0_BRANCH/src/con
trib/org/apache/commons/httpclient/contrib/proxy/PluginProxyUtil.java $
 * $Revision: 1.5 $
 * $Date: 2007/01/18 11:11:27 $
 * 
 * ====================================================================
 * 
 *  Copyright 1999-2004 The Apache Software Foundation
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * 
 * ====================================================================
 */
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

jets3t : Java Extra-Tasty S3 Toolkit (for Amazon S3 online storage service)
This is a java.net project, see https://jets3t.dev.java.net/

* Copyright 2008 James Murty
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

/*
 * $HeadURL:
https://svn.apache.org/repos/asf/jakarta/commons/proper/httpclient/branches/HTTPCLIENT_3_0_BRANCH/src/con
trib/org/apache/commons/httpclient/ contrib/proxy/ProxyDetectionException.java $ *
 * $Revision: 1.1 $ *
 * $Date: 2006/11/10 10:36:45 $ *
 *
 * ==================================================================== *

 * Copyright 1999-2004 The Apache Software Foundation *
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); *
 * you may not use this file except in compliance with the License. *
 * You may obtain a copy of the License at *
 *
 * http://www.apache.org/licenses/LICENSE-2.0 *
 *
 * Unless required by applicable law or agreed to in writing, software *
 * distributed under the License is distributed on an "AS IS" BASIS, *
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. *
 * See the License for the specific language governing permissions and *
 * limitations under the License. *
 *
 * ==================================================================== *
 *
 * This software consists of voluntary contributions made by many *
 * individuals on behalf of the Apache Software Foundation. For more *
 * information on the Apache Software Foundation, please see *
 * <http://www.apache.org/>. *
 *
 */
No license file was found, but licenses were detected in source scan.

/*
 * $HeadURL:
https://svn.apache.org/repos/asf/jakarta/commons/proper/httpclient/branches/HTTPCLIENT_3_0_BRANCH/src/con
trib/org/apache/commons/httpclient/contrib/proxy/PluginProxyTestApplet.java $
 * $Revision: 1.1 $
 * $Date: 2006/11/10 10:36:45 $
 *
 * ====================================================================
 *
 *  Copyright 1999-2004 The Apache Software Foundation
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * ====================================================================
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Apache Software Foundation. For more
 * information on the Apache Software Foundation, please see
 *
 */

1.2263 typetools 0.5.0
1.2263.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
1.2265 intl-messageformat 2.2.0 2.2.0

1.2265.1 Available under license:

Copyright 2013 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------------
Pluralization rules built from
https://github.com/papandreou/node-cldr
which is licensed under the BSD license and has the following license:

Copyright (c) 2012, Andreas Lind Petersen
All rights reserved.

See the following for more details:
https://github.com/papandreou/node-cldr/blob/master/LICENSE

1.2266 progress 1.1.8
1.2266.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2267 jsreport-browser-client 2.2.2 2.2.2
1.2267.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2268 jsreport-docx 2.9.0 2.9.0
1.2268.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2269 hbase-shaded-jersey 3.4.1
1.2269.1 Available under license:

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

**Angular JS, v1.6.6**
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

**aopalliance Version 1**
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

**Bean Validation API 2.0.2**
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

**Hibernate Validator CDI, 6.1.2.Final**
* License: Apache License, 2.0
* Project: https://beanvalidation.org/
* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

**Bootstrap v3.3.7**
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

**Google Guava Version 18.0**
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javajoinject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 8.0
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.
W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/

# Notice for Jersey Core Common module
This content is produced and maintained by the Eclipse Jersey project.

* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright: (C) 2009 The Guava Authors

JSR-166 Extension - JEP 266
* License: Creative Commons 1.0 (CC0)
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program.

http://creativecommons.org/publicdomain/zero/1.0/
# Eclipse Public License - v 2.0
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient’s receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS"
BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF
TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR
PURPOSE. Each Recipient is solely responsible for determining the
appropriateness of using and distributing the Program and assumes all
risks associated with its exercise of rights under this Agreement,
including but not limited to the risks and costs of program errors,
compliance with applicable laws, damage to or loss of data, programs
or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS
SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this Agreement, and without further
action by the parties hereto, such provision shall be reformed to the
minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity
(including a cross-claim or counterclaim in a lawsuit) alleging that the
Program itself (excluding combinations of the Program with other software
or hardware) infringes such Recipient's patent(s), then such Recipient's
rights granted under Section 2(b) shall terminate as of the date such
litigation is filed.

All Recipient's rights under this Agreement shall terminate if it
fails to comply with any of the material terms or conditions of this
Agreement and does not cure such failure in a reasonable period of
time after becoming aware of such noncompliance. If all Recipient's
rights under this Agreement terminate, Recipient agrees to cease use
and distribution of the Program as soon as reasonably practicable.
However, Recipient's obligations under this Agreement and any licenses
granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement,
but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and
any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2270 jsbn 0.1.1 0.1.1

1.2270.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2271 dbf-reader 0.3.0
1.2271.1 Available under license:

MIT License

Copyright (c) 2016 Yellow Feather Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2272 finalhandler 1.1.2 1.1.2

1.2272.1 Available under license:

(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2275 node-extend 2.0.2 2.0.2

1.2275.1 Available under license:
Dual licensed. MIT and GPL2

The MIT License (MIT)

Copyright (c) 2011 Jordan Harband and Ben Lin

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---------------------------------------

Open Source Used In DNAC Platform Arica 10822
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

{description}

Copyright (C) {year} {fullname}

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

{signature of Ty Coon}, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.2276 babel-helper-create-class-features-plugin 7.16.0

1.2276.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2277 babel-helper-explode-assignable-expression 6.24.1 6.24.1

1.2277.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2278 has-symbols 1.0.1 1.0.1

1.2278.1 Available under license:
MIT License

Copyright (c) 2016 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2279 @babel/helper-explode-assignable-expression 7.16.0

1.2279.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2280 sshpk 1.16.1 1.16.1
1.2280.1 Available under license:

Copyright Joyent, Inc. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.2281 symbol-observable 1.2.0 1.2.0
1.2281.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Copyright (c) Ben Lesh <ben@benlesh.com>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2282 httpcomponents-client 4.5.13
1.2282.1 Available under license:
Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

========================================================================

This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
version 1.1 or earlier of the License, but not also under the
terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
Software; or

(b) any new file in Source Code Form that contains any Covered
Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.
1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
Licensable by such Contributor to use, reproduce, make available,
modify, display, perform, distribute, and otherwise exploit its
Contributions, either on an unmodified basis, with Modifications, or
as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
for sale, have made, import, and otherwise transfer either its
Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License.
Notwithstanding Section 2.1(b) above, no patent license is granted by a
Contributor:

(a) for any code that a Contributor has removed from Covered Software;
or

(b) for infringements caused by: (i) Your and any other third party's
modifications of Covered Software, or (ii) the combination of its
Contributions with other software (except as part of its Contributor
Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form
If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************

6. Disclaimer of Warranty

Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or
*  statutory, including, without limitation, warranties that the  *
*  Covered Software is free of defects, merchantable, fit for a  *
*  particular purpose or non-infringing. The entire risk as to the  *
*  quality and performance of the Covered Software is with You.  *
*  Should any Covered Software prove defective in any respect, You  *
*  (not any Contributor) assume the cost of any necessary servicing,  *
*  repair, or correction. This disclaimer of warranty constitutes an  *
*  essential part of this License. No use of any Covered Software is  *
*  authorized under this License except under this disclaimer.  *
*  *
************************************************************************
************************************************************************
************************************************************************

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort  *
(including negligence), contract, or otherwise, shall any  *
Contributor, or anyone who distributes Covered Software as  *
permitted above, be liable to You for any direct, indirect,  *
special, incidental, or consequential damages of any character  *
including, without limitation, damages for lost profits, loss of  *
goodwill, work stoppage, computer failure or malfunction, or any  *
and all other commercial damages or losses, even if such party  *
shall have been informed of the possibility of such damages. This  *
limitation of liability shall not apply to liability for death or  *
personal injury resulting from such party’s negligence to the  *
extent applicable law prohibits such limitation. Some  *
jurisdictions do not allow the exclusion or limitation of  *
incidental or consequential damages, so this exclusion and  *
limitation may not apply to You.  *

************************************************************************

8. Litigation

-------------

Any litigation relating to this License may be brought only in the  *
courts of a jurisdiction where the defendant maintains its principal  *
place of business and such litigation shall be governed by laws of that  *
jurisdiction, without reference to its conflict-of-law provisions.  *
Nothing in this Section shall prevent a party's ability to bring  *
cross-claims or counter-claims.

9. Miscellaneous

-------------
This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

---------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

-------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.2283 parquet-common 1.8.1

1.2283.1 Available under license:

Apache Parquet Common
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
   distribution, then any Derivative Works that You distribute must
   include a readable copy of the attribution notices contained
   within such NOTICE file, excluding those notices that do not
   pertain to any part of the Derivative Works, in at least one
   of the following places: within a NOTICE text file distributed
   as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
   within a display generated by the Derivative Works, if and
   wherever such third-party notices normally appear. The contents
   of the NOTICE file are for informational purposes only and
   do not modify the License. You may add Your own attribution
   notices within Derivative Works that You distribute, alongside
   or as an addendum to the NOTICE text from the Work, provided
   that such additional attribution notices cannot be construed
   as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2284 jwa 1.1.6 1.1.6
1.2284.1 Available under license :

Copyright (c) 2013 Brian J. Brennan

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the
Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2285 slf4j-log4j 1.7.32
1.2285.1 Available under license :

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 *
*/

1.2286 semver-diff 3.1.1
1.2286.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.2287 js-tokens 3.0.2
1.2287.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.2288 scala 2.11.0
1.2288.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2012 FuseSource, Inc.
 */
1.2289 react-ace 5.10.0 5.10.0
1.2289.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 James Hrisho

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
1.2290 parquet-jackson 1.10.0
1.2290.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product depends on Jackson and includes it in this binary artifact. Jackson is a high-performance JSON processor.
1.2291 istanbuljs-schema 0.1.3
1.2291.1 Available under license:
MIT License

Copyright (c) 2019 CFWare, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2292 univocity-parsers 2.8.3
1.2292.1 Available under license:
No license file was found, but licenses were detected in source scan.

/************************************************************
 * Copyright 2015 Univocity Software Pty Ltd
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
************************************************************/
* See the License for the specific language governing permissions and
  * limitations under the License.
  **********************************************************************************************/

Found in path(s):
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_29.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/record/RecordMetaData.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/record/MetaData.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/csv/CsvRoutines.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/fixed/FieldAlignment.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/EnumSelector.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/EnumConversion.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/EnumOptions.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/EnumSelector.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/EnumConversion.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Headers.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Header.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/processors/ConcurrentRowProcessorTest.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/processors/OutputValueSwitch.java
  * /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
z/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/processors/ConcurrentRowProcessorTest.java

Open Source Used In DNAC Platform Arica 10858
Open Source Used In DNAC Platform Arica 10859
No license file was found, but licenses were detected in source scan.

/*******************************************************************************
* Copyright 2014 Univocity Software Pty Ltd
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*******************************************************************************/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_31.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/InputAnalysisProcess.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/AbstractInputValueSwitch.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_35.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/NoopRowProcessorErrorHandler.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/LineReader.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/RowWriterProcessorSwitch.java

Open Source Used In DNAC Platform Aric  10861
Open Source Used In DNAC Platform Arica

10862
Open Source Used In DNAC Platform Arica 10867

```
/opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/common/processor/ObjectColumnProcessorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/BatchedColumnProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/tsv/TsvParserTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/common/processor/AnnotatedBeanProcessorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/fixed/FixedWidthWriter.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/CharInputReader.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/examples/CsvParserExamples.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/helpers/FieldMapping.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/fields/ExcludeIndexSelectorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/NullStringConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/ByteConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/NumberConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/DateTimeConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/FormatConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/TextFieldConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/EnumConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/CompositeConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectColumnProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowWriterProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/AnnotatedColumnProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/IndexedColumnProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/RowWriterProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ColumnProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ.gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ColumnBaseProcessor.java
```
Open Source Used In DNAC Platform Arica 10868

gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/CommonParserSettings.java
* /opt/ws_local/PERMITS_SQL/1092334798_1060713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/Context.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/Context.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowWriterProcessorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/ColumnReader.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/DefaultCharInputReader.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/DefaultCharInputReader.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/WriterCharAppender.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/csv/CsvFormat.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/RowPlacement.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/AbstractListProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/input/concurrent/CharBucket.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/helpers/AnnotationHelper.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/FormattedBigDecimalConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/fields/FieldSelector.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/MasterDetailRecord.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/ObjectRowWriterProcessorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_10160713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/uniVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/tsv/TsvWriterSettings.java
No license file was found, but licenses were detected in source scan.

/*******************************************************************************
* Copyright 2017 Univocity Software Pty Ltd
*Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
*You may obtain a copy of the License at

Open Source Used In DNAC Platform Arica 10869
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
***********************************************************************

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_183.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/iterators/RecordIterator.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/helpers/TransformedHeader.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/HeaderTransformer.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Nested.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/processor/core/CompositeProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_154.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_198.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_197.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Nested.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Nested.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_176.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_191.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_75.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_138.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/iterators/RowIterator.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/iterators/RecordIterator.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_149.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_177.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
Open Source Used In DNAC Platform Arica  10871
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/CompositeRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_180.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_182.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_178.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_146.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/IterableResult.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/Iterables/ParserIterator.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_194.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/helpers/MethodFilter.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotaions/Iterables/ParserIterator.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/Iterables/ParserIterator.java
No license file was found, but licenses were detected in source scan.

/*******************************************************************************
 * Copyright 2016 Univocity Software Pty Ltd
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 ******************************************************************************/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/examples/samples/CsvSearchExample.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_128.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/csv/Csv.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/examples/FixedWidthWriterExamples.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_109.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/csv/UnescapedQuoteHandling.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/support/Ticket_7.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-
Open Source Used In DNAC Platform Arica  10873
No license file was found, but licenses were detected in source scan.

/**
 * ****************************************************************************
 * Copyright 2014 Univocity Software Pty Ltd
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on
an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions and limitations under the License.
* ******************************************************************************
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/TextParsingException.java
No license file was found, but licenses were detected in source scan.

/*******************************************************************************
* Copyright 2015 Univocity Software Pty Ltd
* <p>
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* <p>
* http://www.apache.org/licenses/LICENSE-2.0
* <p>
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*******************************************************************************/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_24.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 Univocity Software Pty Ltd
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-
gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/record/RecordImpl.java

Open Source Used In DNAC Platform Arica 10877
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2015. Univocity Software Pty Ltd
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/beans/PropertyWrapper.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/beans/BeanHelper.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 Univocity Software Pty Ltd
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/record/RecordFactory.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/fields/FieldEnumSelector.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2016 Univocity Software Pty Ltd.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/common/input/WriterCharAppenderTest.java

No license file was found, but licenses were detected in source scan.

/*******************************************************************************
* Copyright 2019 Univocity Software Pty Ltd
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*******************************************************************************/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_309.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_315.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_328.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-tar-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2013 uniVocity Software Pty Ltd. All rights reserved.
 * This file is subject to the terms and conditions defined in file
 * 'LICENSE.txt', which is part of this source code package.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
zg/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/fields/AbstractColumnMapping.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
zg/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/fields/AbstractColumnMapping.java

Apache License, Version 2.0 - The Apache Software Foundation
Apache License<br>
<a href="http://www.apache.org/licenses/">http://www.apache.org/licenses/</a>
Subject to the terms and conditions of

Open Source Used In DNAC Platform Arica 10880
Subject to the terms and conditions of attribution notices from the Source form of the Work, include a readable copy of the attribution notices contained do not modify the License. You may add Your own attribution that such additional attribution notices cannot be construed This License does not grant permission to use the trade

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/LICENSE-2.0.html
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2015. Univocity Software Pty Ltd
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/CustomMatcher.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/ConversionProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/common/processor/MultiBeanListProcessorTest.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/MultiBeanRowProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/MultiBeanProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/MultiBeanListProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-gz/univocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/...
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2018. Univocity Software Pty Ltd
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_255.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/annotations/Validate.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/conversions/ValidatedConversion.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/ParsingContext.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/ParsingContextWrapper.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/AbstractMultiBeanListProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_37.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/test/java/com/univocity/parsers/issues/github/Github_39.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/AbstractMultiBeanProcessor.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-targ-
gz/unVocity-univocity-parsers-a19fc0e/src/main/java/com/univocity/parsers/common/processor/core/AbstractMultiBeanRowProcessor.java
No license file was found, but licenses were detected in source scan.


Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-target/univocity-univocity-parsers-a19fcede/src/main/java/com/univocity/parsers/common/processor/core/AbstractProcessorSwitch.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-target/univocity-univocity-parsers-a19fcede/src/main/java/com/univocity/parsers/common/processor/core/RowProcessorSwitch.java
* /opt/ws_local/PERMITS_SQL/1092334798_1600713115.55/0/univocity-univocity-parsers-v2-8-3-0-gcaf1775-target/univocity-univocity-parsers-a19fcede/src/main/java/com/univocity/parsers/common/Internal.java

Open Source Used In DNAC Platform Arica 10883
1.2293 client-sessions 0.8.0 0.8.0

1.2293.1 Available under license:
Mozilla Public License Version 2.0

1. Definitions

1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"

(a) that the initial Contributor has attached the notice described
    in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
    version 1.1 or earlier of the License, but not also under the
    terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.
1.9. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications" means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License" means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form" means the form of the work preferred for making modifications.

1.14. "You" (or "Your") means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants
Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).
2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work
You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

-----------
5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particularContributor are reinstated (a) provisionally, unless and until suchContributor explicitly and finally terminates Your grants, and (b) on anongoing basis, if such Contributor fails to notify You of thenon-compliance by some reasonable means prior to 60 days after You havecome back into compliance. Moreover, Your grants from a particularContributor are reinstated on an ongoing basis if such Contributornotifies You of the non-compliance by some reasonable means, this is thefirst time You have received notice of non-compliance with this Licensefrom such Contributor, and You become compliant prior to 30 days afterYour receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patentinfringement claim (excluding declaratory judgment actions,counter-claims, and cross-claims) alleging that a Contributor Versiondirectly or indirectly infringes any patent, then the rights granted toYou by any and all Contributors for the Covered Software under Section2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, allend user license agreements (excluding distributors and resellers) whichhave been validly granted by You or Your distributors under this Licenseprior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an    *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************
************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a
10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
1.2294.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2295 extglob 0.3.2
1.2295.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2296 html-escaper 2.0.2

1.2296.1 Available under license:

Copyright (C) 2017-present by Andrea Giammarchi - @WebReflection

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2297 esutils 2.0.2 2.0.2

1.2297.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.2298 prelude-ls 1.1.2

1.2298.1 Available under license:

Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2299 ms 2.1.3

1.2299.1 Available under license:

The MIT License (MIT)

Copyright (c) 2020 Vercel, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2300 set-immediate-shim 1.0.1 1.0.1

1.2300.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2301 write 0.2.1

1.2301.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2302 twill-api 0.6.0-incubating

1.2302.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/internal/DefaultTwillSpecification.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/api/Racks.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/api/ClassAcceptor.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/api/ServiceController.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/api/SecureStore.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/internal/DefaultEventHandlerSpecification.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/api/ServiceAnnouncer.java
* /opt/cola/permits/1137746702_1614121526.26/0/twill-api-0-6-0-incubating-sources-3-
jar/org/apache/twill/internal/RunIds.java
1.2303 nock 11.9.1

1.2303.1 Available under license:

MIT License

Copyright (c) 2011-2019 Pedro Teixeira and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
# Contributors

This list has been generated to thank everyone here for their contribution. Nock wouldn't be what it is without the individuals who've given their time to it. In alphabetical order, here is a list of people who the maintainers think are worthy of being thanked publicly and permanently here.

Aalhad Saraf @saraf https://github.com/saraf
Aaron D Borden @adborden https://github.com/adborden
Aaron Greenwald @aarongreenwald https://github.com/aarongreenwald
Abdul Alkhatib @abdulito https://github.com/abdulito
Abdullah Ali @AbdullahAli https://github.com/AbdullahAli
Abhijeet Singh @abhijeetps https://github.com/abhijeetps
Abimbola Idowu @hisabimbola https://github.com/hisabimbola
Adam @snowmac https://github.com/snowmac
Adam Bretz @arb https://github.com/arb
Adam Cofer @acofer https://github.com/acofer
Adam Ever-Hadani @adamhadani https://github.com/adamhadani
Adam LaFave @lafave https://github.com/lafave
Adam Tyler @AdamTyler https://github.com/AdamTyler
Adam Voss @adamvoss https://github.com/adamvoss
Adara @adarra https://github.com/adarra
Aditya Bhardwaj @purezen https://github.com/purezen
Aditya Gupta @aditya1315 https://github.com/aditya1315
Adrian Mejia @amejiarosario https://github.com/amejiarosario
Adrian Perez @blackxored https://github.com/blackxored
Aidan Feldman @afeld https://github.com/afeld
Akshaya Kumar Sharma @lordakshaya https://github.com/lordakshaya
Alan Clarke @alanclarke https://github.com/alanclarke
Alan Lee @mlalanlee https://github.com/mlalanlee
Alastair Brunton @pyrat https://github.com/pyrat
Alberto Gasparin @albertogasparin https://github.com/albertogasparin
Alberto Pose @pose https://github.com/pose
Aleksandr Barszczewski @alekbarszczewski https://github.com/alekbarszczewski
Aleksandr Gidenko @aleksgidenko https://github.com/aleksgidenko
Aleksi Pekkala @epiphone https://github.com/epiphone
Alex Banks @MoonTahoe https://github.com/MoonTahoe
Alex Dunn @dunn https://github.com/dunn
Alex Gilleran @AlexGilleran https://github.com/AlexGilleran
Alex Layton @awlayton https://github.com/awlayton
Alex Olson @alexkolson https://github.com/alexkolson
Alex Panov @alexpanov https://github.com/alexpanov
Alex Pavlov @macscripter https://github.com/macscripter
Alex Voitau @voitau https://github.com/voitau
Alex Zylman @azylman https://github.com/azylman
Alexander Christiansson @alexanderchr https://github.com/alexanderchr
Alexander Rusakov @arusakov https://github.com/arusakov
Alexander Simmerl @xla https://github.com/xla
Alexander Tesfamichael @alextes https://github.com/alextes
Open Source Used In DNAC Platform Arica 10904

Duke Jones @dukejones https://github.com/dukejones
Dustin Deus @StarpTech https://github.com/StarpTech
Dustin J. Mitchell @djmitche https://github.com/djmitche
Dylan Millikin @PommeVerte https://github.com/PommeVerte
Dvid Szakllas @dszakallas https://github.com/dszakallas
E W @walshe https://github.com/walshe
E. Itah @eitah https://github.com/eitah
Ed @anotheredward https://github.com/anotheredward
Ed Moore @eddiemoore https://github.com/eddiemoore
Edan Schwartz @eschwartz https://github.com/eschwartz
Eddie Antonio Santos @eddieantonio https://github.com/eddieantonio
Edo Rivai @edorivai https://github.com/edorivai
Eduard Bondarenko @eduardbcom https://github.com/eduardbcom
Elnur Abdurrahimov @elnur https://github.com/elnur
Emelia Smith @ThisIsMissEm https://github.com/ThisIsMissEm
Emily M Coco @emilycoco https://github.com/emilycoco
Enzo Hernan Ncolorich @dkenzo https://github.com/dkenzo
Eric @calidion https://github.com/calidion
Eric Adamski @ericadamski https://github.com/ericadamski
Eric Hulburd @erichulburd https://github.com/erichulburd
Eric Saboia @ericasaboia https://github.com/ericasaboia
Eric Skogen @audionerd https://github.com/audionerd
Erik Rothwell @bacchusrx https://github.com/bacchusrx
Erik W @technicallyerik https://github.com/technicallyerik
Erol Akarsu @eakarsu https://github.com/eakarsu
Esa-Matti Suuronen @epeli https://github.com/epeli
Espen Hovlandsdal @rexxars https://github.com/rexxars
Esteban @invernizzie https://github.com/invernizzie
Ethan Arrowood @Ethan-Arwood https://github.com/Ethan-Arwood
Ethan Garofolo @juanpaco https://github.com/juanpaco
Eugene Krevenets @hyzhak https://github.com/hyzhak
Evgenii Ponomarev @Elergy https://github.com/Elergy
Evgeny Zislis @kesor https://github.com/kesor
Evo O. E. @maziey93 https://github.com/maziey93
Fabiano Frana @fabiano https://github.com/fabiano
Facundo Olano @facundoolano https://github.com/facundoolano
Faiq Raza @faiq https://github.com/faiq
Farid Nouri Neshat @alFReD-NSH https://github.com/alFReD-NSH
Felipe Dornelas @felipead https://github.com/felipead
Felipe Rocha @felipellrocha https://github.com/felipellrocha
Ferriel Melarpis @FerrielMelarpis https://github.com/FerrielMelarpis
Filip Skokan @panva https://github.com/panva
Filipe Deschamps @filipedeschamps https://github.com/filipedeschamps
Francesc Rosas @frosas https://github.com/frosas
Francesco Negri @dhinus https://github.com/dhinus
Herman @hermansb https://github.com/hermansb
Hidenari Nozaki @ghiden https://github.com/ghiden
Hirad Yazdanpanah @hiradyazdan https://github.com/hiradyazdan
Hugo Dures @hugoduraes https://github.com/hugoduraes
Hutson Betts @hbetts https://github.com/hbetts
Huu Da Tran @danosaure https://github.com/danosaure
Hyunju Choi @blackpost38 https://github.com/blackpost38
Ian Walker-Sperber @ianwsperber https://github.com/ianwsperber
Ian Young @iangreenleaf https://github.com/iangreenleaf
Igor Gali @igalic https://github.com/igalic
Igor Plekhan @ipelekhan https://github.com/ipelekhan
Ihor Poplavskyi @Poplava https://github.com/Poplava
Ilya Antipenko @aivus https://github.com/aivus
Ilya Sotov @lsotov https://github.com/lsotov
Imran Chaudhry @imran-uk https://github.com/imran-uk
Ingwar Wirjawan @rawgni https://github.com/rawgni
Ioan Lucut @ioanlucut https://github.com/ioanlucut
Ioannis Poulakas @giannis https://github.com/giannis
Islam Sharabash @ibash https://github.com/ibash
Ivan Erceg @ierceg https://github.com/ierceg
Ivan Poluyanov @poluyanov https://github.com/poluyanov
Izaak Rogan @izaakroban https://github.com/izaakroban
Izik Lisbon @izikl https://github.com/izikl
Jack Franklin @jackfranklin https://github.com/jackfranklin
Jack Stevens @jpstevens https://github.com/jpstevens
Jacob Wejendorp @wejendorp https://github.com/wejendorp
Jake @samouri https://github.com/samouri
Jake Craige @jakecraige https://github.com/jakecraige
Jake Pruitt @jakepruitt https://github.com/jakepruitt
Jakub Holy @jakubholynet https://github.com/jakubholynet
James Fitness @jamesfitness https://github.com/jamesfitness
James Harrison Fisher @jameshf https://github.com/jameshf
James Herdman @jherdman https://github.com/jherdman
James Pace @pacey https://github.com/pacey
James Talmage @jamestal https://github.com/jamestal
Jamie Jennings @jamiemjennings https://github.com/jamiemjennings
Jamison Dance @jergason https://github.com/jergason
Jan Lehnardt @janl https://github.com/janl
Jan Potoms @Janpot https://github.com/Janpot
Jane Nguyen @pichimochi https://github.com/pichimochi
Jared Klopper @optical https://github.com/optical
Jarid Margolin @jaridmargolin https://github.com/jaridmargolin
Jason Avinger @jasonav https://github.com/jasonav
Jason Fischl @jfischl https://github.com/jfischl
Jason Galea @lecstor https://github.com/lecstor
Jason Kuhrt @jasonkuhrt https://github.com/jasonkuhrt
Jason Plumb @breedx2 https://github.com/breedx2
Jasper Kuperus @jasperkuperus https://github.com/jasperkuperus
Jay Chae @chaekit https://github.com/chaekit
Jeff @visualjeff https://github.com/visualjeff
Jeff Burn @jeffora https://github.com/jeffora
Jeff Dickey @jdxcode https://github.com/jdxcode
Jeff Lee @jeffomatic https://github.com/jeffomatic
Jeffrey Charles @jeffcharles https://github.com/jeffcharles
Jeffrey Jagoda @jagoda https://github.com/jagoda
Jehan @jtremback https://github.com/jtremback
Jens Claes @entropitor https://github.com/entropitor
Jens Peter Secher @jsecher https://github.com/jsecher
JeongHoon Byun (aka Outsider) @outsideris https://github.com/outsideris
Jeremiah Lee @jeremiahlee https://github.com/jeremiahlee
Jeremy Sik @jeremysik https://github.com/jeremysik
Jerome Gravel-Niquet @jeromegn https://github.com/jeromegn
Jess Legans-Combarro @piranna https://github.com/piranna
Jezeniel Zapanta @jezeniel https://github.com/jezeniel
Jhuliano Skittberg Moreno @jhuliano https://github.com/jhuliano
Jianzhong Chen @cjjzcsyx https://github.com/cjjzcsyx
Jiayi Hu @jiayihu https://github.com/jiayihu
Jim Brusstar @jimbru https://github.com/jimbru
Jinwoo Lee @jinwoo https://github.com/jinwoo
Jiri Spac @capaj https://github.com/capaj
Joaquim Serafin @joaquimserafim https://github.com/joaquimserafim
Joe Becher @drazisil https://github.com/drazisil
Joe Kent @itsjoekent https://github.com/itsjoekent
Joe Norman @joseph-norman https://github.com/joseph-norman
Joe Smith @Yasumoto https://github.com/Yasumoto
Johan Borestad @borestad https://github.com/borestad
Johan Nordberg @jnordberg https://github.com/jnordberg
Johannes Jrg Schmidt @jo https://github.com/jo
Johannes Zellner @nebulade https://github.com/nebulade
John @jhnlsn https://github.com/jhnlsn
John Duhamel @jjduhamel https://github.com/jjduhamel
John Dzak @jdzak https://github.com/jdzak
John Mathis @JohnDMathis https://github.com/JohnDMathis
John T @digitaltsai https://github.com/digitaltsai
John-David Dalton @jdalton https://github.com/jdalton
Jon B @PizzaAficionado https://github.com/PizzaAficionado
Jon Jaques @jonjaques https://github.com/jonjaques
Jon Skulski @jskulski https://github.com/jskulski
Jonas Lilja @jilija https://github.com/jilija
Jonas Scheffner @jscheffner https://github.com/jscheffner
Jonathan Bergknoff @jbergknoff https://github.com/jbergknoff
Jonathan Glock @glockjt https://github.com/glockjt
Jonathan Lau @jonahlav https://github.com/jonahlav
Jonathan Otto @jotto https://github.com/jotto
Jonathan Petitcolas @jpeticolas https://github.com/jpeticolas
Jonathan Rainey @Tivoli https://github.com/Tivoli
Jonathon Herbert @jonathonherbert https://github.com/jonathonherbert
Joni Salmi @josalmi https://github.com/josalmi
Jordan Harbarg @ljharb https://github.com/ljharb
Joseph Cellaars @bcallaars https://github.com/bcallaars
Josh Crowther @jshcrowthe https://github.com/jshcrowthe
Josh Marchello @jmarchello https://github.com/jmarchello
Josh McMillan @mcmillan https://github.com/mcmillan
Joshua Acheson @joshacheson https://github.com/joshacheson
Joshua Holbrook @jfholbrook https://github.com/jfholbrook
Josip Delic @delijati https://github.com/delijati
Jos F. Romaniello @jfromaniello https://github.com/jfromaniello
Joo Ferreira @jmnsf https://github.com/jmnsf
Juan Cruz Viotti @jviotti https://github.com/jviotti
Julian Hille @julianhille https://github.com/julianhille
Juraci de Lima Vieira Neto @Juraci https://github.com/Juraci
Justin @justincy https://github.com/justincy
Justin Helmer @justinhelmer https://github.com/justinhelmer
Justin Marrington @HowlingEverett https://github.com/HowlingEverett
Jrme Avoustin @rehia https://github.com/rehia
Jrg Henning @joerx https://github.com/joerx
Kai @hyalkaf https://github.com/hyalkaf
Kai Schwarz @papakai https://github.com/papakai
Kaiwalya Kher @knkher https://github.com/knkher
Kalashnikov Igor @silentroach https://github.com/silentroach
Kamal Marhubi @kamalmarhubi https://github.com/kamalmarhubi
Karina Jain @sacredMonster https://github.com/sacredMonster
Karl Atkinson @karlatkinson https://github.com/karlatkinson
Karlo Janyst @LKay https://github.com/LKay
Ke Zhu @shawnzhu https://github.com/shawnzhu
Keith Laban @kelaban https://github.com/kelaban
Kelly @kellyrmilligan https://github.com/kellyrmilligan
Ken Ding @choonkending https://github.com/choonkending
Ken Sheedlo @ksheedlo https://github.com/ksheedlo
Kevin Bloch (@codingthat) @codingthat https://github.com/codingthat
Kevin Burke @kevinburke https://github.com/kevinburke
Kevin Burke @kevinburkesh https://github.com/kevinburkesh
Kevin Hodges @kevinhodges https://github.com/kevinhodges
Kevin Ingersoll @holic https://github.com/holic
Kevin Locke @kevinoid https://github.com/kevinoid
Kevin McDermott @bigkevmcd https://github.com/bigkevmcd
Kevin Smith @ksmith97 https://github.com/ksmith97
Kevin Whitaker @kwhitaker https://github.com/kwhitaker
Kevin Woo @kevinawoo https://github.com/kevinawoo
Keyvan Fatehi @kfatehi https://github.com/kfatehi
Khizar @khizar https://github.com/khizar
Kilian Ciuffolo @kilianc https://github.com/kilianc
Kimmo Brunfeldt @kimmobrunfeldt https://github.com/kimmobrunfeldt
Kirill Enykeev @enykeev https://github.com/enykeev
Komran Rashidov @krashidov https://github.com/krashidov
Konstantin Koss @KonstantinKo https://github.com/KonstantinKo
Kostas @vrinek https://github.com/vrinek
Krishna Rajendran @blazzy https://github.com/blazzy
Kun Yan @kunyan https://github.com/kunyan
Kyle White @chainlink https://github.com/chainlink
Kvin Bernard-Allies @BAKFR https://github.com/BAKFR
LAU Thierry @lauterry https://github.com/lauterry
Lars Haler @LarsHassler https://github.com/LarsHassler
Lars Nystrm @larsnystrom https://github.com/larsnystrom
Leo Iannacone @LeoIannacone https://github.com/LeoIannacone
Leo Liang @aleung https://github.com/aleung
Leo Soto @leosoto https://github.com/leosoto
Leon Rinkel @leonrinkel https://github.com/leonrinkel
Leonardo Couto @wallacyyy https://github.com/wallacyyy
Leonid Blinov @lblinovs https://github.com/lblinovs
Levi Wheatcroft @leviwheatcroft https://github.com/leviwheatcroft
Liam Morley @carpeliam https://github.com/carpeliam
Linus Unnebck @LinusU https://github.com/LinusU
Lior Brauer @liorbrauer https://github.com/liorbrauer
Little Gnome @LittleGnome https://github.com/LittleGnome
Logan Allred @redbugz https://github.com/redbugz
Logan Tegman @ltegman https://github.com/ltegman
Lohann Paterno Coutinho Ferreira @Lohann https://github.com/Lohann
Loc Mahieu @LoicMahieu https://github.com/LoicMahieu
Luan Nico @luanpotter https://github.com/luanpotter
Lucas Feliciano @lucasfeliciano https://github.com/lucasfeliciano
Lucas Lago @lucaslago https://github.com/lucaslago
Luigi Pinca @lpinca https://github.com/lpinca
Luis Del guila @delaguilaluis https://github.com/delaguilaluis
Luke Childs @lukechilds https://github.com/lukechilds
Luke Nimtz @protometa https://github.com/protometa
Luu Ninh @ninhxuanluu https://github.com/ninhxuanluu
MJ @maheshjag https://github.com/maheshjag
Mac Angell @mac- https://github.com/mac-
Maciek Sakrejda @uhoh-itsmaciek https://github.com/uhoh-itsmaciek
Madan Raj @mannadan03 https://github.com/mannadan03
Malcolm Rebughini @malcolmrebughini https://github.com/malcolmrebughini
Manan Jadhav @CurosMJ https://github.com/CurosMJ
Maor Hayoun @maorhayoun https://github.com/maorhayoun
Marc Boscher @marcboscher https://github.com/marcboscher
Marco @marco-c https://github.com/marco-c
Marco Nascimento (Coi) @Coooi https://github.com/Coooi
Mario Mendes @hyprstack https://github.com/hyprstack
Mario Mol @mariohmol https://github.com/mariohmol
Mario Pareja @mpareja https://github.com/mpareja
Mark Meyer @mark-meyer https://github.com/mark-meyer
Mark van Cuijk @phedny https://github.com/phedny
Mickal Tricot @mickaeltr https://github.com/mickaeltr
Miguel Espinoza @purefan https://github.com/purefan
Miguel Isidoro @misidoro https://github.com/misidoro
Miha Rebernik @mihar https://github.com/mihar
Mikael Arneborn @marneborn https://github.com/marneborn
Mike Chen @mhchen https://github.com/mhchen
Mike Emery @MikeEmery https://github.com/MikeEmery
Mike James @export-mike https://github.com/export-mike
Mike Weinstein @mreinstein https://github.com/mreinstein
Mikela @glassresistor https://github.com/glassresistor
Mikey Powers @mvpowers https://github.com/mvpowers
Mikko Koponen @mtkopone https://github.com/mtkopone
Minh Son Nguyen @nguymin4 https://github.com/nguymin4
Mohammad Salim @MSalim https://github.com/MSalim
Moo Yeol Prescott Lee @mooyoul https://github.com/mooyoul
Naga Pavan Kumar Kunisetty @nagapavan https://github.com/nagapavan
Nate Wang @supnate https://github.com/supnate
Nathan @naw https://github.com/naw
Nathan Friedly @nfriedly https://github.com/nfriedly
Nathan Houle @ndhoule https://github.com/ndhoule
Nathan Richards @LAD500 https://github.com/LAD500
Neeraj @smurfpandey https://github.com/smurfpandey
Nic Haynes @nicinabox https://github.com/nicinabox
Nicholas Calugar @SocalNick https://github.com/SocalNick
Nick Howes @halfninja https://github.com/halfninja
Nick Stefan @NickStefan https://github.com/NickStefan
Nicky Jay @niksajanjc https://github.com/niksajanjc
Nico Schlmer @nschloe https://github.com/nschloe
Nicolas Del Valle @ndelvalle https://github.com/ndelvalle
Nicolas Froidure @nfroidure https://github.com/nfroidure
Nicolas Lacasse @nlacasse https://github.com/nlacasse
Nicols Bevacqua @bevacqua https://github.com/bevacqua
Niels Krijger @nielskrijger https://github.com/nielskrijger
Nikita Vasilyevski @Nigrimmist https://github.com/Nigrimmist
Nikolaus Piccolotto @prayerslayer https://github.com/prayerslayer
Nils Burger @nilsbunger https://github.com/nilsbunger
Nils Knappmeier @knknap https://github.com/nknapp
Nock Bot @nockbot https://github.com/nockbot
Nordes Mnard-Lamarre @Nordes https://github.com/Nordes
Nuno Job @dscape https://github.com/dscape
Nuno Sousa @nunofgs https://github.com/nunofgs
Oleg @omakoleg https://github.com/omakoleg
Oliver Wong @owiber https://github.com/owiber
Olivier Louvignes @mgcrea https://github.com/mgcrea
Olivier Tassinari @oliviertassinari https://github.com/oliviertassinari
Omar Khan @omarkhan https://github.com/omarkhan
Ondej dych @zdychacek https://github.com/zdychacek
Operations Research Engineering Software+ @ORESoftware https://github.com/ORESoftware
Richard Lay @richardlay https://github.com/richardlay
Richard Littauer @RichardLitt https://github.com/RichardLitt
Richard Meijer @richmeij https://github.com/richmeij
Richard Scarrott @richardscarrott https://github.com/richardscarrott
Richard Watkins @RichardWatkins1 https://github.com/RichardWatkins1
Rico Sta. Cruz @rstacruz https://github.com/rstacruz
Rob Ballou @robballoul https://github.com/robballou
Rob Calcroft @robcalcroft https://github.com/robcalcroft
Rob Forsythe @robforsythe https://github.com/robforsythe
Rob Stevenson @thisdotrob https://github.com/thisdotrob
Rob Wu @Rob--W https://github.com/Rob--W
Robert Hurst @RobertWHurst https://github.com/RobertWHurst
Robert Simpson @rsimp https://github.com/rsimp
Robin Bobbitt @robinbobbitt https://github.com/robinbobbitt
Roch Devost @rochdev https://github.com/rochdev
Rocky Assad @CodisRedding https://github.com/CodisRedding
Rod Howard @rodhoward https://github.com/rodhoward
Rodrigo Gomes da Silva @rodrigogs https://github.com/rodrigogs
Rohit Kalkur @rovolution https://github.com/rovolution
Rolf Strijdhorst @rolfst https://github.com/rolfst
Romain @rprieto https://github.com/rprieto
Roman @moltar https://github.com/moltar
Roman Hotsiy @RomanGotsiy https://github.com/RomanGotsiy
Roman Pearah @neverfox https://github.com/neverfox
Roman Shtyyman @defunctzombie https://github.com/defunctzombie
Ronal Jouchet @ronjouch https://github.com/ronjouch
Ronen Amiel @ronami https://github.com/ronami
Rong Sen Ng @motss https://github.com/motss
Rory Dent @Thebigbignooby https://github.com/Thebigbignooby
Roy Choo @roychoo https://github.com/roychoo
Ruben Verborgh @RubenVerborgh https://github.com/RubenVerborgh
Rudy Jahchan @rudyjahchan https://github.com/rudyjahchan
Rui Marinho @ruimarinho https://github.com/ruimarinho
Rui Quelhas @ruiquelhas https://github.com/ruiquelhas
Ruimou Xu @noperative https://github.com/noperative
Russell Dempsey @SgtPooki https://github.com/SgtPooki
Ryan Juve @ryanjeu-porch https://github.com/ryanjeu-porch
Ryan Lewis @ryanmurakami https://github.com/ryanmurakami
Ryan Zander @Exide https://github.com/Exide
Ryan Zec @ryanzeec https://github.com/ryanzeec
Ryunosuke Sato @tricknotes https://github.com/tricknotes
Rmi Bges @Overdrivr https://github.com/Overdrivr
Rmi Jarasson @ArTiSTiX https://github.com/ArTiSTiX
Rmy Boulanouar @DblK https://github.com/DblK
Rmy HUBSUCHER @Natim https://github.com/Natim
SARUNYHOT SUVANNACHOTI @chenka https://github.com/chenka
Sahat Yalkabov @sahat https://github.com/sahat
Sam Saccone @samccone https://github.com/samccone
Sam Vloegerhs @samvloeberghs https://github.com/samvloeberghs
Samuel Roldan @sam3k https://github.com/sam3k
Samuel Thompson @samuel11 https://github.com/samuel11
Sarah Hayman @serahhh https://github.com/serahhh
Sarah Vessels @cheshire137 https://github.com/cheshire137
Saran Siriphanphon @deoxygen0n https://github.com/deoxygen0n
Sascha Drees @perfusorius https://github.com/perfusorius
Sad Tayebi @SaidTayebi https://github.com/SaidTayebi
Scott Tesoriere @scottk https://github.com/scottk
Sean @seanmccintyre https://github.com/seanmccintyre
Sean Massa @endangeredmassa https://github.com/EndangeredMassa
Sean McLeann @oceanswave https://github.com/Oceanswave
Sebastian Durandeu @sdurandeu https://github.com/sdurandeu
Sebastian Thiel @Byron-TW https://github.com/Byron-TW
Seth Miller @four43 https://github.com/four43
Seth Westphal @westy92 https://github.com/westy92
Sethen Maleno @sethen https://github.com/sethen
Sen Hayes @SeanHayes https://github.com/SeanHayes
Shahar Talmi @shahata https://github.com/shahata
Shai @shai32 https://github.com/shai32
Sharikov Vladislav @sharikovvladislav https://github.com/sharikovvladislav
Shota @shotaK https://github.com/shotaK
Shrey @shreychaturvedi123 https://github.com/shreychaturvedi123
Shu Pengfei @stormslovely https://github.com/stormslovely
Shuan Wang @swang https://github.com/swang
Siddhartha Dabral @sidabs https://github.com/sidabs
Silvio Rainoldi @ianaz https://github.com/ianaz
Simen Bekkhus @SimenB https://github.com/SimenB
Simon Brewster @ssbrewster https://github.com/ssbrewster
Simon Elvery @drzax https://github.com/drzax
Simon Legg @leggsimon https://github.com/leggsimon
Simon Taylor @s-taylor https://github.com/s-taylor
Simon Turvey @serenitus https://github.com/serenitus
Sindre Sorhus @sindresorhus https://github.com/sindresorhus
Soroush Pour @soroushp https://github.com/soroushp
Spencer @spalger https://github.com/spalger
Srdjan Cencic @srdimir https://github.com/srdimir
Stas Slesarev @ceoworks https://github.com/ceoworks
Stephen Cresswell @cressie176 https://github.com/cressie176
Stephen Matthew Daviess @stevematdavies https://github.com/stevematdavies
Stephen Solka @trashhalo https://github.com/trashhalo
Stephen Sugden @grncdr https://github.com/grncdr
Stephen Woods @saw https://github.com/saw
Steve Gentile @sgentile https://github.com/sgentile
Steve Oliveira @sloveira-nascent https://github.com/sloveira-nascent
Steven Olsen @solsend2l https://github.com/solsend2l
Steven Oxley @xonev https://github.com/xonev
Steven Vachon @stevenvachon https://github.com/stevenvachon
Stian Grytyr @stiang https://github.com/stiang
Stuart Dotson @sdotson https://github.com/sdotson
Sunil Lulla @sunil-lulla https://github.com/sunil-lulla
Suraj Byanju @surajbyanju https://github.com/surajbyanju
Sushant @sushantdhiman https://github.com/sushantdhiman
Sven Lito @svlito https://github.com/svlito
Swift @theycallmeswift https://github.com/theycallmeswift
Szymon Przybylski @sprzybylski https://github.com/sprzybylski
Tanner Vanman @THoisington https://github.com/THoisington
Taylor Bockman @angrygoats https://github.com/angrygoats
Taylor Dawson @taylorjdawson https://github.com/taylorjdawson
Taylor Everding @taylor1791 https://github.com/taylor1791
Tedde Lundgren @tedeh https://github.com/tede
Terin Stock @terinjokes https://github.com/terinjokes
TheFive @TheFive https://github.com/TheFive
Theo Gravity @theogravity https://github.com/theogravity
Thiebaud Thomas @thomastiebaud https://github.com/thomastiebaud
Thierry Schellenbach @tschellenbach https://github.com/tschellenbach
Thomas Alexander @tomalex0 https://github.com/tomalex0
Thomas Heymann @cyberthom https://github.com/cyberthom
Thomas Lahr @tjahr https://github.com/tjahr
Thomas Parisot @oncletom https://github.com/oncletom
Thomas Rix @rixth https://github.com/rixth
Thomas Shafer @trshafer https://github.com/trshafer
Thomas Watson @watson https://github.com/watson
Tiago @tiagocpontesp https://github.com/tiagocpontesp
Tiago Ribeiro @fixe https://github.com/fixe
Tiagojsag @tiagojsag https://github.com/tiagojsag
Tim Jonischkat @timjonischkat https://github.com/timjonischkat
Tim Kindberg @timkindberg https://github.com/timkindberg
Tim Perry @pimterry https://github.com/pimterry
Tim Robinson @timrobinson https://github.com/timrobinson
Tim Rogers @timrogers https://github.com/timrogers
Tim Savery @timsavery https://github.com/timsavery
Tim Yager @creativetim https://github.com/creativetim
Tim Zaitsev @tzaitsev https://github.com/tzaitsev
Tobias Kaupat @Niondir https://github.com/Niondir
Tom @tconroy https://github.com/tconroy
Tom @traverse https://github.com/traverse
Tom Beddard @subblue https://github.com/subblue
Tom Dunlap @motevets https://github.com/motevets
Tom Spencer @fiznool https://github.com/fiznool
Tomasz Janczuk @tjanczuk https://github.com/tjanczuk
Trevor L @landau https://github.com/landau
Trevor Livingston @tlivings https://github.com/tlivings
Trevor Muraro @TrevorMuraro https://github.com/TrevorMuraro
Trym Skaar @trym https://github.com/trym
Tudor Gergely @tudorgergely https://github.com/tudorgergely
Tushar Mathur @tusharmath https://github.com/tusharmath
Tyler @tyrw https://github.com/tyrw
Tyler Benziger @tybenz https://github.com/tybenz
Tyler Childs @tylerchilds https://github.com/tylerchilds
Tyler Renelle @lefniere https://github.com/lefniere
Vahe Hovhannisyan @vhpoet https://github.com/vhpoet
Vasiliy Yorokin @vyorkin https://github.com/vyorkin
Venkat @Venkat-18 https://github.com/Venkat-18
Vincent Quigley @vquigley https://github.com/vquigley
Vincent Voyer @vvo https://github.com/vvo
Vinh Bachsy @vinh0604 https://github.com/vinh0604
Vitalii Honcharenko @vithonch https://github.com/vithonch
Vitalij Kudresov @kudresov https://github.com/kudresov
Vitor Arins @vitorarins https://github.com/vitorarins
Vojtech Novak @vonovak https://github.com/vonovak
Vsevolod Strukchinsky @floatdrop https://github.com/floatdrop
Wes Johnson @sterlingwes https://github.com/sterlingwes
Will Myers @griffinmyers https://github.com/griffinmyers
Will Prater @wprater https://github.com/wprater
Will Simons @bigredwill https://github.com/bigredwill
Willson Mock @fay-jai https://github.com/fay-jai
Winson Tsang @winsontsang https://github.com/winsontsang
Xavier Zhou @xavierchou https://github.com/xavierchou
Yann Odeyer @yodeyer https://github.com/yodeyer
Yaroslav @mrself https://github.com/mrself
Yin Rong @yinrong https://github.com/yinrong
Yves Richard @whyvez https://github.com/whyvez
Zbyszek Tenerowicz @naugtur https://github.com/naugtur
Zeke Sikelianos @zeke https://github.com/zeke
Zhanzhao (Deo) Liang @DeoLeung https://github.com/DeoLeung
Zlatko @zladuric https://github.com/zladuric
Zoli Kahan @Zolmeister https://github.com/Zolmeister
Zuri Pabin @zuripabon https://github.com/zuripabon
chad king @kingchad https://github.com/kingchad
chen @koroshi https://github.com/koroshi
cjroebuck @cjroebuck https://github.com/cjroebuck
cv @carolynnvu https://github.com/carolynnvu
edA-qa mort-ora-y @em-cliqz https://github.com/em-cliqz
eddie.chen @chenghung https://github.com/chenghung
hems.io @hems https://github.com/hems
huanghaidiyang @huanghaidiyang https://github.com/huanghaidiyang
jerishi @qqcloud https://github.com/qqcloud
jess @monkeywithacupcake https://github.com/monkeywithacupcake
marco @ayxos https://github.com/ayxos
mscdex @mscdex https://github.com/mscdex
n30n0v @n30n0v https://github.com/n30n0v
narendra @reddynr https://github.com/reddynr
@4no0p https://github.com/4no0p
@Alex0007 https://github.com/Alex0007
@AlexRRR https://github.com/AlexRRR
@AndersAstrand https://github.com/AndersAstrand
@Hyjaz https://github.com/Hyjaz
@JuHwon https://github.com/JuHwon
@MarshallRJ https://github.com/MarshallRJ
@NRaf https://github.com/NRaf
@Nagarjuna-S https://github.com/Nagarjuna-S
@Niggler https://github.com/Niggler
@Nohbidy https://github.com/Nohbidy
@Nutelac https://github.com/Nutelac
@PatSmuk360 https://github.com/PatSmuk360
@UndeadBaneGitHub https://github.com/UndeadBaneGitHub
@abhishekkn-optimus https://github.com/abhishekkn-optimus
@accooper-accusoft https://github.com/accooper-accusoft
@alexscheelmeyer https://github.com/alexscheelmeyer
@andremrodrigues https://github.com/andremrodrigues
@andrewaustin https://github.com/andrewaustin
@apptous-seb https://github.com/apptous-seb
@ashishbhr2k https://github.com/ashishbhr2k
@bhar629 https://github.com/bhar629
@bialesdaniel https://github.com/bialesdaniel
@bitgal https://github.com/bitgal
@blumus https://github.com/blumus
@brian-poncho https://github.com/brian-poncho
@chitsuhein https://github.com/chitsuhein
@clydehunt76 https://github.com/clydehunt76
@congdcit https://github.com/congdcit
@craigcosmo https://github.com/craigcosmo
@ddragos https://github.com/ddragos
@deepaktiwari-1987 https://github.com/deepaktiwari-1987
@devang-kredx https://github.com/devang-kredx
@divo-ie https://github.com/divo-ie
@dragoplut https://github.com/dragoplut
@dunse https://github.com/dunse
@dynamitesushi https://github.com/dynamitesushi
@esatterwhite https://github.com/esatterwhite
@ewoutmeyns https://github.com/ewoutmeyns
@fent https://github.com/fent
@flushentitypacket https://github.com/flushentitypacket
@galenus https://github.com/galenus
@getlittletech https://github.com/getlittletech
@gmatroskin https://github.com/gmatroskin
@greenkeeper null
@handane123 https://github.com/handane123
@hellboy81 https://github.com/hellboy81
@hermano360 https://github.com/hermano360
@ianmacl https://github.com/ianmacl
1.2304 txw2 2.3.1

1.2304.1 Available under license:

Found license 'General Public License 2.0' in Copyright (c) 2005-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in " Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in Copyright (c) 2005-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Open Source Used In DNAC Platform Arica 10919
Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License // This class is in the Public Domain. NO WARRANTY! * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>'

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License // NO WARRANTY! This class is in the public domain.'

1.2305 on-finished 2.3.0 2.3.0
1.2305.1 Available under license :

(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2014 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
1.2306 libproxy 0.4.11-5ubuntu1

1.2306.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.
For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY. TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.2307 jetty-http-spi 9.3.27.v20190418

1.2308 is-callable 1.1.4

1.2308.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2309 shallow-clone 3.0.1

1.2309.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2310 codemodel 2.6
1.2311 snake-yaml 1.27
1.2311.1 Available under license:

Found license 'GNU Lesser General Public License' in ' // This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

Found license 'Eclipse Public License 1.0' in ' // This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bfd-licence.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

1.2312 calcite-cassandra 1.21.0-drill-r2
1.2312.1 Available under license:

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to you under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# Database of contributors to Apache Calcite.
# Pages such as developer.md use this data.
# List must be sorted by first name, last name.
#
- name: Alan Gates
  emeritus: 2018/05/04
  apacheId: gates
  githubId: alanfgates
  org: Hortonworks
  role: PMC
- name: Aman Sinha
  apacheId: amansinha
githubId: amansinha100
org: MapR
role: PMC
- name: Andrei Sereda
  apacheId: sereda
  githubId: asereda-gs
  org:
  role: Committer
- name: Ashutosh Chauhan
  apacheId: hashutosh
  githubId: ashutoshec
  org: Hortonworks
  role: PMC
- name: Chris Baynes
  apacheId: cbaynes
  githubId: chris-baynes
  org: Contiamo
  role: Committer
- name: Christian Beikov
  apacheId: cbeikov
  githubId: beikov
  org: Blazebit
  role: Committer
- name: Chunwei Lei
  apacheId: chunwei
  githubId: chunweilei
  org: Alibaba
  role: Committer
- name: Danny Chan
  apacheId: danny0405
  githubId: danny0405
  org: Alibaba
  role: Committer
- name: Edmon Begoli
  apacheId: ebegoli
  githubId: ebegoli
  org: Oak Ridge National Laboratory
  role: Committer
- name: Francis Chuang
  apacheId: francischuang
  githubId: F21
  org: Boostport
  role: PMC Chair
- name: Gian Merlino
  apacheId: gian
  githubId: gianm
  org: Imply
  role: Committer
- name: Haisheng Yuan
  apacheId: hyuan
  githubId: hsyuan
  org: Alibaba
  role: Committer

- name: Hongze Zhang
  apacheId: hongze
  githubId: zhizheplaye
  org: Tencent
  role: Committer

- name: James R. Taylor
  apacheId: jamestaylor
  githubId: JamesRTaylor
  org: Salesforce
  role: PMC

- name: Jacques Nadeau
  apacheId: jacques
  githubId: jacques-n
  org: Dremio
  role: PMC

- name: Jess Balint
  apacheId: jbalint
  githubId: jbalint
  org: Stardog
  role: Committer

- name: Jess Camacho Rodriguez
  apacheId: jcamacho
  githubId: jcamachor
  org: Hortonworks
  role: PMC

- name: Jinfeng Ni
  apacheId: jni
  githubId: jinfengni
  org: MapR
  role: PMC

- name: John Pullokkaran
  apacheId: jpullokk
  githubId: jpullokkaran
  org: Hortonworks
  role: PMC

- name: Josh Elser
  apacheId: elserj
  githubId: joshelser
  org: Hortonworks
  role: PMC

- name: Julian Hyde
  apacheId: jhyde
  githubId: julianhyde
org: Looker
role: PMC
homepage: http://people.apache.org/~jhyde
- name: Kevin Liew
  apacheId: kliew
githubId: kliewkliw
  org:
  role: Committer
- name: Kevin Risden
  apacheId: krisden
githubId: risdenk
  org:
  role: PMC
- name: Laurent Goujon
  apacheId: laurent
githubId: laurentgo
  org: Dremio
  role: PMC
- name: Maryann Xue
  apacheId: maryannxue
githubId: maryannxue
  org: Intel
  role: PMC
- name: Michael Mior
  apacheId: mmior
githubId: michaelmior
  org: Rochester Institute of Technology
  role: PMC
  homepage: https://michael.mior.ca/
- name: Milinda Pathirage
  apacheId: milinda
githubId: milinda
  org: Indiana University
  role: Committer
  homepage: http://milinda.pathirage.org/
- name: MinJi Kim
  apacheId: minji
githubId: minji-kim
  org: Oracle
  role: Committer
- name: Nick Dimiduk
  apacheId: ndimiduk
githubId: ndimiduk
  org:
  role: PMC
  homepage: http://www.n10k.com
- name: Nishant Bangarwa
  apacheId: nishant
githubId: nishantmonu51
org: Hortonworks
role: Committer
- name: Ruben Quesada Lopez
  apacheId: rubenql
  githubId: rubenada
  org: TIBCO
  role: Committer
- name: Sergey Nuyanzin
  apacheId: snuyanzin
  githubId: snuyanzin
  org: EPAM
  role: Committer
- name: Shuyi Chen
  apacheId: shuyichen
  githubId: suez1224
  org: Uber
  role: Committer
- name: Slim Bouguerra
  apacheId: bslim
  githubId: b-slim
  org: Hortonworks
  role: Committer
- name: Stamatis Zampetakis
  apacheId: zabetak
  githubId: zabetak
  org: TIBCO
  role: PMC
- name: Steven Noels
  apacheId: stevenn
  githubId: stevenn
  org: NGData
  role: PMC
- name: Ted Dunning
  apacheId: tdunning
  githubId: tdunning
  org: MapR
  role: PMC
  avatar: https://mapr.com/blog/author/ted-dunning/assets/tdunning-panama.jpg
- name: Vladimir Sitnikov
  apacheId: vladimirsitnikov
  githubId: vlsi
  org: NetCracker
  role: PMC
- name: Volodymyr Vysotskyi
  apacheId: volodymyr
  githubId: vvysotskyi
  org:
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-----------------------------------------------------------------------

APACHE CALCITE SUBCOMPONENTS:

The Apache Calcite project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

-----------------------------------------------------------------------

The MIT License
The Apache Calcite project bundles the following files under the MIT License:

- site
  Parts of the web site generated by Jekyll (http://jekyllrb.com/)
  Copyright (c) 2008-2015 Tom Preston-Werner
- site/_sass/_font-awesome.scss
  Font-awesome css files v4.1.0 (http://fortawesome.github.io/Font-Awesome/)
  Copyright (c) 2013 Dave Gandy
- site/_sass/_normalize.scss
  normalize.css v3.0.2 | git.io/normalize
  Copyright (c) Nicolas Gallagher and Jonathan Neal
- site/_sass/_gridism.scss
  Gridism: A simple, responsive, and handy CSS grid by @cobyism
  https://github.com/cobyism/gridism
  Copyright (c) 2013 Coby Chapple
- site/js/html5shiv.min.js
  HTML5 Shiv 3.7.2 | @afarkas @jdalton @jon_neal @rem
- site/js/respond.min.js
  Respond.js v1.4.2: min/max-width media query polyfill
  Copyright 2013 Scott Jehl

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Open Font License

The Apache Calcite project bundles the following fonts under the SIL Open Font License (OFL) - http://scripts.sil.org/OFL/
Apache Calcite
Copyright 2012-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product is based on source code originally developed
by DynamoBI Corporation, LucidEra Inc., SQLstream Inc. and others
under the auspices of the Eigenbase Foundation
and released as the LucidDB project.

The web site includes files generated by Jekyll.

1.2313 argparse 1.0.10
1.2313.1 Available under license :
(The MIT License)

Copyright (C) 2012 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2314 sprintf-js 1.0.3
1.2314.1 Available under license :
MIT License

Copyright (c) 2019 Johannes Tegnr / Jitesoft
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2315 node-promise 0.5.13 0.5.13

1.2315.1 Available under license:

No license file was found, but licenses were detected in source scan.

MIT License.

The node-promise project provides a complete promise implementation. Promises provide a clean separation of concerns between asynchronous behavior and the interface so asynchronous functions can be called without callbacks, and callback interaction can be done on the generic promise interface. The node-promise package provides just a promise implementation, however, https://github.com/kriszyp/promised-io is recommended for more complete promise-based IO functionality. The promised-io includes the promise implementation from node-promise, as well as wrappers around Node's filesystem and other system I/O APIs for consistent promise-based interaction.

The node-promise module features a promise implementation with:

* Chainable promises
* Promises throw errors if an error handler is not provided
* CommonJS promise proposal [1] compliant
* Immutable once fulfilled to reduce possible side-effects
* Promises can be used securely (as separate resolver/promise pairs in ocap situations)
* Backwards compatibility where possible (addCallback, addErrback, emitSuccess, and emitError should still behave as expected)

Utility functions, including:
* when() - Normalization of sync (normal values) and async (promises)
* all() - Create a promise that accumulate multiple concurrent promises (failed promises resolve to Error objects)
* allOrNone() - Ditto, but the first promise to fail causes the composition to fail as well
* first() - Find the first promise to be fulfilled in a group of promises
* seq() - Sequentially execute a set of promise returning functions
* delay() - Returns a promise that is fulfilled after a given amount of time
* execute() - Executes a function that takes a callback and returns a promise (thank you Benjamin Thomas for providing this)

Much of this is adapted from Tyler Close's ref_send and Kris Kowal's work on promises.

Some quick examples from test-promise.js (again, it is recommended that you use http://github.com/kriszyp/promised-io for file and other I/O interaction):

```javascript
util = require('util');
var fs = require('./fs-promise');

// open a file and read it
fs.open("fs-promise.js", process.O_RDONLY).then(function(fd){
  return fs.read(fd, 4096);
}).then(function(args){
  util.puts(args[0]); // print the contents of the file
});

// does the same thing
fs.readFile("fs-promise.js").addCallback(util.puts);
```

A default Promise constructor can be used to create a self-resolving deferred/promise:

```javascript
var Promise = require("node-promise").Promise;
var promise = new Promise();
asyncOperation(function(){
  promise.resolve("successful result");
});
promise -> given to the consumer
```

A consumer can use the promise:

```javascript
promise.then(function(result){
  ... when the action is complete this is executed ...
},
function(error){
  ... executed when the promise fails
});
```

Alternately, a provider can create a deferred and resolve it when it completes an action. The deferred object a promise object that provides a separation of consumer and producer to protect promises from being fulfilled by untrusted code.
```javascript
var defer = require("node-promise").defer;
var deferred = defer();
asyncOperation(function(){
    deferred.resolve("succeful result");
});
deferred.promise -> given to the consumer

Another way that a consumer can use promises:

var when = require("node-promise").when;
when(promise,function(result){
    ... when the action is complete this is executed ...
},
function(error){
    ... executed when the promise fails
});

More examples:

function printFirstAndLast(itemsDeferred){
    findFirst(itemsDeferred).then(util.puts);
    findLast(itemsDeferred).then(util.puts);
}
function findFirst(itemsDeferred){
    return itemsDeferred.then(function(items){
        return items[0];
    });
}
function findLast(itemsDeferred){
    return itemsDeferred.then(function(items){
        return items[items.length];
    });
}

And now you can do:

printFirstAndLast(someAsyncFunction());

The workhorse function of this library is the "when" function, which provides a means for normalizing interaction with values and functions that may be a normal synchronous value, or may be a promise (asynchronously fulfilled). The when() function takes a value that may be a promise or a normal value for the first function, and when the value is ready executes the function provided as the second argument (immediately in the case of a non-promise normal value). The value returned from when() is the result of the execution of the provided function, and returns a promise if provided a promise or synchronously returns a normal value if provided a non-promise value. This makes it easy to "chain" computations together. This allows us to write code that is agnostic to sync/async interfaces:

var when = require("node-promise").when;
```
function printFirstAndLast(items){
  // print the first and last item
  when(findFirst(items), util.puts);
  when(findLast(items), util.puts);
}

function findFirst(items){
  // return the first item
  return when(items, function(items){
    return items[0];
  });
}

function findLast(items){
  // return the last item
  return when(items, function(items){
    return items[items.length - 1];
  });
}

Now we can do:

  > printFirstAndLast([1,2,3,4,5]);
  1
  5

And we can also provide asynchronous promise:

  var promise = new process.Promise();
  > printFirstAndLast(promise);

  (nothing printed yet)

  > promise.emitSuccess([2,4,6,8,10]);
  2
  10

The "all" function is intended to provide a means for waiting for the completion of an array of promises. The "all" function should be passed an array of promises, and it returns an promise that is fulfilled once all the promises in the array are fulfilled. The returned promise's resolved value will be an array with the resolved values of all of the promises in the passed in array.

The "first" function is intended to provide a means for waiting for the completion of the first promise in an array of promises to be fulfilled. The "first" function should be passed an array of promises, and it returns an promise that is fulfilled once the first promise in the array is fulfilled. The returned promise's resolved value will be the resolved value of the first fulfilled promise.
No license file was found, but licenses were detected in source scan.

// // MIT License

Found in path(s):
* /opt/cola/permits/1025855972_1636567465.07/0/node-promise-0-5-13-tgz/package/promise.js

1.2316 underscore 1.13.0-2 1.13.0-2
1.2316.1 Available under license:

Copyright (c) 2009-2020 Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2317 mute-stream 0.0.5
1.2317.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2318 chokidar 3.3.1 3.3.1
1.2318.1 Available under license:
The MIT License (MIT)

Copyright (c) 2012-2019 Paul Miller (https://paulmillr.com), Elan Shanker

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the Software), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2319 cliui 4.1.0 4.1.0
1.2319.1 Available under license:
Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software
for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice
appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE
LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2320 babel-plugin-transform-es2015-computed-properties 6.24.1 6.24.1
1.2320.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2321 jsreport-chrome-pdf 1.10.0 1.10.0
1.2321.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at runtime a copy of the Library already present on the user's computer.

Open Source Used In DNAC Platform Arica 10953
system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.
If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2322 xalan 2.7.2
1.2322.1 Available under license:
SUN PUBLIC LICENSE Version 1.0

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof and corresponding documentation released with the source code.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated documentation, interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1 The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) notwithstanding Section 2.2(b) above, no patent license is
granted: 1) for any code that Contributor has deleted from the
Contributor Version; 2) separate from the Contributor Version; 3) for
infringements caused by: i) third party modifications of Contributor
Version or ii) the combination of Modifications made by that
Contributor with other software (except as part of the Contributor
Version) or other devices; or 4) under Patent Claims infringed by
Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are
governed by the terms of this License, including without limitation
Section 2.2. The Source Code version of Covered Code may be
distributed only under the terms of this License or a future version
of this License released under Section 6.1, and You must include a
copy of this License with every copy of the Source Code You
distribute. You may not offer or impose any terms on any Source Code
version that alters or restricts the applicable version of this
License or the recipients’ rights hereunder. However, You may include
an additional document offering the additional rights described in
Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be
made available in Source Code form under the terms of this License
either on the same media as an Executable version or via an accepted
Electronic Distribution Mechanism to anyone to whom you made an
Executable version available; and if made available via Electronic
Distribution Mechanism, must remain available for at least twelve (12)
months after the date it initially became available, or at least six
(6) months after a subsequent version of that particular Modification
has been made available to such recipients. You are responsible for
ensuring that the Source Code version remains available even if the
Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a
file documenting the changes You made to create that Covered Code and
the date of any change. You must include a prominent statement that
the Modification is derived, directly or indirectly, from Original
Code provided by the Initial Developer and including the name of the
Initial Developer in (a) the Source Code, and (b) in any notice in an
Executable version or related documentation in which You describe the
origin or ownership of the Covered Code.

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface ("API") and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than
any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to
understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Sun Microsystems, Inc. ("Sun") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Sun. No one other than Sun has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must: (a) rename Your license so that the phrases "Sun," "Sun Public License," or "SPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Sun Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER
8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant’s Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.
8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.


11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and
expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

Exhibit A -Sun Public License Notice.

The contents of this file are subject to the Sun Public License Version 1.0 (the "License"); you may not use this file except in compliance with the License. A copy of the License is available at http://www.sun.com/

The Original Code is _______________. The Initial Developer of the Original Code is ___________. Portions created by _____ are Copyright (C)_________. All Rights Reserved.

Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the SPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the SPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of
the notices in the Source Code files of the Original Code. You should
use the text of this Exhibit A rather than the text found in the
Original Code Source Code for Your Modifications.]
xml-commons/java/external/LICENSE.dom-documentation.txt $Id: LICENSE.dom-documentation.txt,v 1.2
2005/06/03 22:49:13 mrglavas Exp $

This license came from: http://www.w3.org/Consortium/Legal/copyright-documents-20021231

W3C DOCUMENT LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231

Public documents on the W3C site are provided by the copyright holders under
the following license. By using and/or copying this document, or the W3C
document from which this statement is linked, you (the licensee) agree that
you have read, understood, and will comply with the following terms and
conditions:

Permission to copy, and distribute the contents of this document, or the W3C
document from which this statement is linked, in any medium for any purpose
and without fee or royalty is hereby granted, provided that you include the
following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it
doesn't exist, a notice (hypertext is preferred, but a textual
representation is permitted) of the form: "Copyright [date-of-document]
World Wide Web Consortium, (Massachusetts Institute of Technology,
European Research Consortium for Informatics and Mathematics, Keio
University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be
provided. We request that authorship attribution be provided in any software,
documents, or other items or products that you create pursuant to the
implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted
pursuant to this license. However, if additional requirements (documented in
the Copyright FAQ) are satisfied, the right to create modifications or
derivatives is sometimes granted by the W3C to individuals complying with
those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO
REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT
LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE,
NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005/06/03 22:49:13 $

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Geronimo
Copyright 2003-2006 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache OpenEJB
Copyright 1999-2009 The Apache OpenEJB development community
This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

This license came from: http://www.w3.org/Consortium/Legal/copyright-software-20021231

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.
This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005/06/03 22:49:13 $

========================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License,   ==
==  Version 2.0, in this case for the Apache Xalan Java distribution.  ==
========================================================================

Apache Xalan (Xalan XSLT processor)
Copyright 1999-2014 The Apache Software Foundation

Apache Xalan (Xalan serializer)
Copyright 1999-2012 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

========================================================================
Portions of this software was originally based on the following:
- software copyright (c) 1999-2002, Lotus Development Corporation.,
- software copyright (c) 2001-2002, Sun Microsystems.,
- software copyright (c) 2003, IBM Corporation.,

========================================================================

The binary distribution package (ie. jars, samples and documentation) of this product includes software developed by the following:

- The Apache Software Foundation
  - Xerces Java - see LICENSE.txt
  - JAXP 1.3 APIs - see LICENSE.txt
  - Bytecode Engineering Library - see LICENSE.txt
  - Regular Expression - see LICENSE.txt
- Scott Hudson, Frank Flannery, C. Scott Ananian
  - CUP Parser Generator runtime (javacup\runtime) - see LICENSE.txt

==========================================================================
The source distribution package (ie. all source and tools required to build
Xalan Java) of this product includes software developed by the following:

- The Apache Software Foundation
  - Xerces Java - see LICENSE.txt
  - JAXP 1.3 APIs - see LICENSE.txt
  - Bytecode Engineering Library - see LICENSE.txt
  - Regular Expression - see LICENSE.txt
  - Ant - see LICENSE.txt
  - Stylebook doc tool - see LICENSE.txt

- Elliot Joel Berk and C. Scott Ananian
  - Lexical Analyzer Generator (JLex) - see LICENSE.txt

==========================================================================
Apache Xerces Java
Copyright 1999-2006 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of Apache Xerces Java in xercesImpl.jar and xml-apis.jar
were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

==========================================================================
Apache xml-commons xml-apis (redistribution of xml-apis.jar)

Apache XML Commons

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Open Source Used In DNAC Platform Artica 10977
The license above applies to this Apache Xalan release of:
Xalan-Java 2 - XSLT Processor
Xalan-Java 2 - Serializer

The license above also applies to the jar files
xalan.jar and xslte.jar - Xalan-Java 2 - XSLT Processor from
Source: http://xalan.apache.org/

The license above also applies to the jar file
serializer.jar - Xalan-Java 2 - Serializer
Source: http://xalan.apache.org/
Used by: Xalan-Java 2 and Xerces-Java 2

The license above also applies to the jar file
xercesImpl.jar - Xerces-Java 2 XML Parser.
Source: http://xerces.apache.org/
Used by: Xalan-Java 2

The license above also applies to the jar file
xml-apis.jar - Xerces-Java 2 XML Parser.
Source: http://xerces.apache.org/
Used by: Xalan-Java 2 and release copy of Xerces-Java 2

The following license applies to the included files:
tools/ant.jar
tools/antRun
tools/antRun.bat
Source: http://ant.apache.org/
Used By: Xalan's build process: java/build.xml and test/build.xml

<--------------------------------------------------------------------------------------------------------------------------------->
/*
 * The Apache Software License, Version 1.1
 * http://www.apache.org/licenses/LICENSE-1.1
 * Copyright (C) 1999 The Apache Software Foundation. All rights reserved.
 * Redistibution and use in source and binary forms, with or without modifica-
 * tion, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org)."

Alternatively, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.
The following license, Apache Software License, Version 1.1, applies to the included BCEL.jar from Apache Jakarta (Byte Code Engineering Library).
Source: http://jakarta.apache.org/bcel
Used By: XSLTC component of xml-xalan/java

The following license, Apache Software License, Version 1.1, also applies to the included regexp.jar, jakarta-regexp-1.2.jar from Apache Jakarta.
Source: http://jakarta.apache.org/regexp
Used By: BCEL.jar which is used by XSLTC component of xml-xalan/java

/*
 *
 * Copyright (c) 2001 The Apache Software Foundation. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 *
 * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 *
 * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 * "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
 * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation" and "Apache BCEL" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache", "Apache BCEL", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* ====================================================================

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
*
* >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

The following license applies to the DOM documentation
for the org.w3c.dom.* packages:

<<<<<<<<<<<W3C DOCUMENT LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231
Public documents on the W3C site are provided by the copyright holders
under the following license. By using and/or copying this document,
or the W3C document from which this statement is linked, you (the licensee)
agree that you have read, understood, and will comply with the following
terms and conditions:

Permission to copy, and distribute the contents of this document, or the
W3C document from which this statement is linked, in any medium for any
purpose and without fee or royalty is hereby granted, provided that you include
the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it
doesn't exist, a notice (hypertext is preferred, but a textual representation
is permitted) of the form: "Copyright [Sdate-of-document] World Wide Web
Consortium, (Massachusetts Institute of Technology, European Research

3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

----------------------------------------------------------------------------

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <mailto:site-policy@w3.org>
Last revised by Reagle $Date: 2005-07-19 12:33:09 -0400 (Tue, 19 Jul 2005) $
The following license applies to the DOM software,
for the org.w3c.dom.* packages in jar file xml-apis.jar:

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231
This work (and included software, documentation such as READMEs,
or other related items) is being provided by the copyright holders
under the following license. By obtaining, using and/or copying this
work, you (the licensee) agree that you have read, understood, and will
comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its
documentation, with or without modification, for any purpose and
without fee or royalty is hereby granted, provided that you include
the following on ALL copies of the software and documentation or
portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the
   redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices,
   or terms and conditions. If none exist, the W3C Software Short Notice
   should be included (hypertext is preferred, text is permitted) within
   the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the
date changes were made. (We recommend you provide URIs to the location
   from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE
ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising
or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

____________________________________
This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <mailto:site-policy@w3.org
Last revised by Reagle $Date: 2005-07-19 12:33:09 -0400 (Tue, 19 Jul 2005) $

The following license applies to the SAX software, for the org.xml.sax.* packages in jar file xml-apis.jar:

This module, both source code and documentation, is in the Public Domain, and comes with NO WARRANTY. See http://www.saxproject.org for further information.

The following license applies to the jar file java_cup.jar - LALR Parser Generator for Java(TM).
CUP Parser Generator Copyright Notice, License, and Disclaimer

Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

The following license applies to the jar file runtime.jar - Component of JavaCup: LALR Parser Generator for Java(TM).

Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors
or their employers not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to
this software, including all implied warranties of merchantability
and fitness. In no event shall the authors or their employers be liable
for any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an action
of contract, negligence or other tortious action, arising out of or
in connection with the use or performance of this software.

The following license applies to the JLEX jar file
JLex.jar - A Lexical Analyzer Generator for Java(TM).
Source:  http://www.cs.princeton.edu/~appel/modern/java/JLex
Used By: XSLTC component of xml-xalan/java

Copyright 1996-2000 by Elliot Joel Berk and C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and
that both the copyright notice and this permission notice and
warranty disclaimer appear in supporting documentation, and that the
name of the authors or their employers not be used in advertising or
publicity pertaining to distribution of the software without specific,
written prior permission.

The authors and their employers disclaim all warranties with regard
to this software, including all implied warranties of merchantability and
fitness. In no event shall the authors or their employers be liable for any
special, indirect or consequential damages or any damages whatsoever resulting
from loss of use, data or profits, whether in an action of contract,
negligence or other tortious action, arising out of or in connection
with the use or performance of this software.

Java is a trademark of Sun Microsystems, Inc. References to the Java
programming language in relation to JLex are not meant to imply that
Sun endorses this product.

The following license applies to the jar file
stylebook-1.0-b3_xalan-2.jar - Tool for generating Xalan documentation.
Integrated with Xalan-Java 2 and Xerces 2.
Source: http://svn.apache.org/viewvc/xml/stylebook/
Used by: Xalan-Java 2, Xalan-C++

/*
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 1999 The Apache Software Foundation. All rights
 * reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 *    if any, must include the following acknowledgment:
 *    "This product includes software developed by the
 *    Apache Software Foundation (http://www.apache.org/)."
 *    Alternately, this acknowledgment may appear in the software itself,
 *    if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Xalan", "Xerces", and "Apache Software Foundation" must
 *    not be used to endorse or promote products derived from this
 *    software without prior written permission. For written
 *    permission, please contact apache@apache.org.
 */
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.apache.org. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

This license came from: http://www.megginson.com/SAX/copying.html

However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

This page is now out of date -- see the new SAX site at http://www.saxproject.org/ for more up-to-date releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under
common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions thereof), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made
available in Source Code form and that Source Code form must be distributed only under the terms of this License.
You must include a copy of this License with every copy of the Source Code form of the Covered Software You
distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a
medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You
represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the
Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered
Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial
Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the
applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee
for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However,
you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must
make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone,
and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the
Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in
compliance with the terms of this License and that the license for the Executable form does not attempt to limit or
alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the
Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby
agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible
for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of
the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or
regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this
License. You agree that You alone are responsible for compliance with the United States export administration
regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise
make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising,
directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer
and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be
deemed to constitute any admission of liability.

1.2323 openssl 1.0.2m
1.2323.1 Available under license :

LICENSE ISSUES
==============

The OpenSSL toolkit stays under a double license, i.e. both the conditions of
the OpenSSL License and the original SSLeay license apply to the toolkit.
See below for the actual license texts.

OpenSSL License
-------------

/* ====================================================================
 * Copyright (c) 1998-2019 The OpenSSL Project.  All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 *    software must display the following acknowledgment:
 *    "This product includes software developed by the OpenSSL Project

Open Source Used In DNAC Platform Arica 10994
for use in the OpenSSL Toolkit. (http://www.openssl.org/)

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions

* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young’s, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*    must display the following acknowledgement:
*    "This product includes cryptographic software written by
*     Eric Young (eay@cryptsoft.com)"
*    The word 'cryptographic' can be left out if the routines from the library
*    being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*    the apps directory (application code) you must include an acknowledgement:
*    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG `AS IS` AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you
duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

   b) use the modified Package only within your corporation or organization.

   c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

   a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

   b) accompany the distribution with the machine-readable source of the Package with your modifications.

   c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a
product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End
1.2325 nanomatch 1.2.13 1.2.13

1.2325.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2326 fontbox 2.0.22

1.2326.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

Open Source Used In DNAC Platform Arica 11010
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

CONTRIBUTIONS TO THE ORIGINAL CODEBASE

Apache FontBox is based on contributions made to the original FontBox project:

Copyright (c) 2006-2007, www.fontbox.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of fontbox; nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Lohit-Bengali font (https://pagure.io/lohit):

Copyright 2011-13 Lohit Fonts Project contributors
<http://fedorahosted.org/lohit>

This Font Software is licensed under the SIL Open Font License, Version 1.1. This license is copied below, and is also available with a FAQ at: http://scripts.sil.org/OFL

-----------------------------------------------------------------------------------

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007
-----------------------------------------------------------------------------------

PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The
fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

DEFINITIONS
"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font
Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Apache FontBox
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Based on source code contributed to the original FontBox project.
Copyright (c) 2006-2007, www.fontbox.org

Includes the Script Property (Scripts-10.0.0.txt)
Copyright 2017 Unicode, Inc.

1.2327 yallist 4.0.0
1.2327.1 Available under license :
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2328 axios 0.21.4 0.21.4
1.2328.1 Available under license :
   Copyright (c) 2014-present Matt Zabriskie

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2329 chalk 2.4.2 2.4.2
1.2329.1 Available under license :
   MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2330 ecdsa-sig-formatter 1.0.11 1.0.11

1.2330.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 D2L Corporation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2331 mimic-response 2.1.0 2.1.0

1.2331.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2332 ansi-regex 4.1.0

1.2332.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2333 estraverse 1.9.3

1.2333.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This Confluent Community License Agreement Version 1.0 (the Agreement) sets forth the terms on which Confluent, Inc. (Confluent) makes available certain software made available by Confluent under this Agreement (the Software). BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO SUCH TERMS AND CONDITIONS, YOU MUST NOT USE THE SOFTWARE. IF YOU ARE RECEIVING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY. Licensee means you, an individual, or the entity on whose behalf you are receiving the Software.

1. LICENSE GRANT AND CONDITIONS.

1.1 License. Subject to the terms and conditions of this Agreement, Confluent hereby grants to Licensee a non-exclusive, royalty-free, worldwide, non-transferable, non-sublicenseable license during the term of this Agreement to: (a) use the Software; (b) prepare modifications and derivative works of the Software; (c) distribute the Software (including without limitation in source code or object code form); and (d) reproduce copies of the Software (the License). Licensee is not granted the right to, and Licensee shall not, exercise the License for an Excluded Purpose. For purposes of this Agreement, Excluded Purpose means making available any software-as-a-service, platform-as-a-service, infrastructure-as-a-service or other similar online service that competes with Confluent products or services that provide the Software.

1.2 Conditions. In consideration of the License, Licensees distribution of the Software is subject to the following conditions:

(a) Licensee must cause any Software modified by Licensee to carry prominent notices stating that Licensee modified the Software.

(b) On each Software copy, Licensee shall reproduce and not remove or alter all Confluent or third party copyright or other proprietary notices contained in the Software, and Licensee must provide the notice below with each copy.

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at http://www.confluent.io/confluent-community-license. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF
1.3 Licensee Modifications. Licensee may add its own copyright notices to modifications made by Licensee and may provide additional or different license terms and conditions for use, reproduction, or distribution of Licensee’s modifications. While redistributing the Software or modifications thereof, Licensee may choose to offer, for a fee or free of charge, support, warranty, indemnity, or other obligations. Licensee, and not Confluent, will be responsible for any such obligations.

1.4 No Sublicensing. The License does not include the right to sublicense the Software, however, each recipient to which Licensee provides the Software may exercise the License so long as such recipient agrees to the terms and conditions of this Agreement.

2. TERM AND TERMINATION. This Agreement will continue unless and until earlier terminated as set forth herein. If Licensee breaches any of its conditions or obligations under this Agreement, this Agreement will terminate automatically and the License will terminate automatically and permanently.

3. INTELLECTUAL PROPERTY. As between the parties, Confluent will retain all right, title, and interest in the Software, and all intellectual property rights therein. Confluent hereby reserves all rights not expressly granted to Licensee in this Agreement. Confluent hereby reserves all rights in its trademarks and service marks, and no licenses therein are granted in this Agreement.

4. DISCLAIMER. CONFLUENT HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE.

5. LIMITATION OF LIABILITY. CONFLUENT WILL NOT BE LIABLE FOR ANY DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, LOST PROFITS OR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, INDIRECT, OR DIRECT DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THE FOREGOING SHALL APPLY TO THE EXTENT PERMITTED BY APPLICABLE LAW.

6. GENERAL.

6.1 Governing Law. This Agreement will be governed by and interpreted in accordance with the laws of the state of California, without reference to its conflict of laws principles. If Licensee is located within the United States, all disputes arising out of this Agreement are subject to the exclusive jurisdiction of courts located in Santa Clara County, California, USA. If Licensee is located outside of the United States, any dispute, controversy or claim arising out of or relating to this
Agreement will be referred to and finally determined by arbitration in accordance with the JAMS International Arbitration Rules. The tribunal will consist of one arbitrator. The place of arbitration will be Palo Alto, California. The language to be used in the arbitral proceedings will be English. Judgment upon the award rendered by the arbitrator may be entered in any court having jurisdiction thereof.

6.2 Assignment. Licensee is not authorized to assign its rights under this Agreement to any third party. Confluent may freely assign its rights under this Agreement to any third party.

6.3 Other. This Agreement is the entire agreement between the parties regarding the subject matter hereof. No amendment or modification of this Agreement will be valid or binding upon the parties unless made in writing and signed by the duly authorized representatives of both parties. In the event that any provision, including without limitation any condition, of this Agreement is held to be unenforceable, this Agreement and all licenses and rights granted hereunder will immediately terminate. Waiver by Confluent of a breach of any provision of this Agreement or the failure by Confluent to exercise any right hereunder will not be construed as a waiver of any subsequent breach of that right or as a waiver of any other right.

Apache Kafka
Copyright 2016 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. Java ClassMate library was originally written by Tatu Saloranta (tatu.salaranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

<title>Apache License, Version 2.0</title>
Apache License
Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions

&#x2219; "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

&#x2219; "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled...
object code, generated documentation, and conversions to other media types.</p>
<p>"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).</p>
<p>"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.</p>
<p>"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."</p>
<p>"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.</p>
<p>2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.</p>
<p>3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.</p>
4. **Redistribution**. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- You must give any other recipients of the Work or Derivative Works a copy of this License; and
- You must cause any modified files to carry prominent notices stating that You changed the files; and
- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions**. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This copy of Jackson JSON processor annotations is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://opensource.org/licenses/mit-license.php">here</a>.</p>
</body></html>

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: schema-registry
Source: https://github.com/confluentinc/schema-registry

Files: *
Copyright: 2018 Confluent, Inc.
License: Apache-2

Files: core/*
Copyright: 2015 Confluent, Inc.
License: Confluent Community License

License: Confluent Community License
Licensed under the Confluent Community License; you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.confluent.io/confluent-community-license

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

License: Apache-2
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

On Debian systems, the Apache 2.0 license can be found in /usr/share/common-licenses/Apache-2.0.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This distribution has a binary dependency on jersey, which is available under the CDDL License as described below.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL - Version 1.1)

1. Definitions.
1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium commonly used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.
4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.
6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The following commands were used to generate license and notice files. Replace <VERSION> with the Schema Registry version, <SRC_PATH> with the path to the Schema Registry source directory, and <LICENSE_TOOL_PATH> with the path of the license tool.

```
cd <SRC_PATH>
mvn package -DskipTests
mkdir /tmp/jars
mkdir /tmp/overrides


cd <LICENSE_TOOL_PATH>
./bin/run_license_job.bash -i /tmp/jars -l <SRC_PATH>/licenses -n <SRC_PATH>/notices -h <SRC_PATH>/licenses-and-notices.html -o /tmp/overrides
```

Apache Avro
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html lang="en-US">
<head>
<title>GlassFish Dual License Header and License Notice GPL v2 and CDDL</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<style type="text/css">
li {display: block;}
</style>
</head>

<body leftmargin="0" topmargin="0" marginheight="0" marginwidth="0" bgcolor="#ffffff">

<h3>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0</h3>

<ul>
<li><p><b>1. Definitions.</b></p>
</li>
</ul>
1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. &ldquo;License&rdquo; means this document.

1.8. &ldquo;Licensable&rdquo; means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. &ldquo;Modifications&rdquo; means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

- under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

- under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.
2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- under intellectual property rights (other than patent or trademark) licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
- under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
- The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
- Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: 
(i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or
<p>trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.</p>

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to
create a new license for Your Original Software, You may create and use a modified version of this License if You:
(a)&nbsp;rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b)&nbsp;otherwise make it clear that the license contains terms which differ from this License.
</p>
</li>
</ul>

<li>
<p><b>5. DISCLAIMER OF WARRANTY.</b></p>

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN &ldquo;AS IS&rdquo; BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
</p>
</li>

<li>
<p><b>6. TERMINATION.</b></p>

<ul>
<li>
<p><b>6.1.</b> This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
</p>
</li>
<li>
<p><b>6.2.</b> If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer
or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant" alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

Open Source Used In DNAC Platform Arica 11059
8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.
10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original
licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
</p>
</li>

</li>
<p>
<b>7.</b> If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
</p>
<p>7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
</p>

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
</p>
<p>
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may
add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the limitation
as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new
versions of the General Public License from time to time. Such new versions
will be similar in spirit to the present version, but may differ in detail
to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any later
version", you have the option of following the terms and conditions either
of that version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of this License,
you may choose any version ever published by the Free Software Foundation.

If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free Software
Foundation, write to the Free Software Foundation; we sometimes make exceptions
for this. Our decision will be guided by the two goals of preserving the
free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE

Open Source Used In DNAC Platform Arics 11068
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library.
you modify this library, you may extend this exception to your version of
the library, but you are not obligated to do so. If you do not wish to do
so, delete this exception statement from your version.
</p>
</li>
</ul>

<p><strong>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1</strong></p>

<p>1. Definitions.</p>

<blockquote>1.1. &ldquo;Contributor&rdquo; means each individual or entity that creates or contributes to the
creation of Modifications.</blockquote>

<blockquote>1.2. &ldquo;Contributor Version&rdquo; means the combination of the Original Software, prior
Modifications used by a Contributor (if any), and the Modifications made by that particular
Contributor.</blockquote>

<blockquote>1.3. &ldquo;Covered Software&rdquo; means (a) the Original Software, or (b) Modifications, or (c)
the combination of files containing Original Software with files containing Modifications, in each case including
portions thereof.</blockquote>

<blockquote>1.4. &ldquo;Executable&rdquo; means the Covered Software in any form other than Source
Code.</blockquote>

<blockquote>1.5. &ldquo;Initial Developer&rdquo; means the individual or entity that first makes Original
Software available under this License.</blockquote>

<blockquote>1.6. &ldquo;Larger Work&rdquo; means a work which combines Covered Software or portions
thereof with code not governed by the terms of this License.</blockquote>

<blockquote>1.7. &ldquo;License&rdquo; means this document.</blockquote>

<blockquote>1.8. &ldquo;Licensable&rdquo; means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed
herein.</blockquote>

<blockquote>1.9. &ldquo;Modifications&rdquo; means the Source Code and Executable form of any of the
following:
<blockquote>A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;</blockquote>

<blockquote>B. Any new file that contains any part of the Original Software or previous Modification;

<blockquote>C. Any new file that is contributed or otherwise made available under the terms of this License.</blockquote>

<blockquote>1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.</blockquote>

<blockquote>1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.</blockquote>

<blockquote>1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</blockquote>

<blockquote>1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</blockquote>

<p>2. License Grants.</p>

<blockquote>2.1. The Initial Developer Grant.</blockquote>

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).</blockquote>

<blockquote>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.</blockquote>

<blockquote>(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.</blockquote>

<blockquote>2.2. Contributor Grant.</blockquote>

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor
Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.
4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment.
or license.</p><blockquote><p>6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.</p></blockquote><p>7. LIMITATION OF LIABILITY.</p><blockquote><p>UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.</p></blockquote><p>8. U.S. GOVERNMENT END USERS.</p><blockquote><p>The Covered Software is a &ldquo;commercial item,&rdquo; as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of &ldquo;commercial computer software&rdquo; (as that term is defined at 48 C.F.R. §252.227-7014(a)(1)) and &ldquo;commercial computer software documentation&rdquo; as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.</p></blockquote><p>9. MISCELLANEOUS.</p><blockquote><p>This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.</p></blockquote><p>10. RESPONSIBILITY FOR CLAIMS.</p><blockquote><p>As between Initial Developer and the Contributors, each party is responsible for claims and damages arising,
directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer
and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be
deeded to constitute any admission of liability. </p>
</blockquote>
<hr />
<p>NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)</p>
<p>The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-
of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts
of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara
County, California. </p>
<p><br />
<br />
</p>
<h1><strong>The GNU General Public License (GPL) Version 2, June 1991</strong></h1>
<p>Copyright (C) 1989, 1991 Free Software Foundation, Inc.<br />
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA</p>
<p>Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. </p>
<p><strong>Preamble</strong></p>
<p>The licenses for most software are designed to take away your freedom to share and change it. By contrast, the
GNU General Public License is intended to guarantee your freedom to share and change free software--to make
sure the software is free for all its users. This General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to using it. (Some other Free Software
Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your
programs, too. </p>
<p>When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are
designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if
you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces
of it in new free programs; and that you know you can do these things. </p>
<p>To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to
surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
software, or if you modify it. </p>
<p>For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the
recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And
you must show them these terms so they know their rights. </p>
<p>We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives
you legal permission to copy, distribute and/or modify the software. </p>
<p>Also, for each author's protection and ours, we want to make certain that everyone understands that there is no
warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to
know that what they have is not the original, so that any problems introduced by others will not reflect on the
original authors' reputations. </p>
<p>Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that
redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary.
To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at
all. </p>
<p>The precise terms and conditions for copying, distribution and modification follow. </p>
<p><strong>TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND
This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The \texttt{Program}, below, refers to any such program or work, and a \texttt{work based on the Program} means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term \texttt{modification}. Each licensee is addressed as \texttt{you}.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be
distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, <blockquote>b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or, <blockquote>c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and any later version, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) &lt;year&gt; &lt;name of author&gt;

Open Source Used In DNAC Platform Arica 11080
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a copyright disclaimer; for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

CERTAIN SOURCE FILES DISTRIBUTED BY ORACLE ARE SUBJECT TO THE FOLLOWING clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the Classpath exception as provided by Oracle in the License file that accompanied this code." exception as provided by Oracle in the License file that accompanied this code.

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
This copy of Jackson JSON processor is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"
The project is licensed under the Confluent Community License, except for client libs, which is under the Apache 2.0 license.

See LICENSE file in each subfolder for detailed license agreement.

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).
The Apache Software Foundation uses various licenses to distribute software and documentation, to accept regular contributions from individuals and corporations, and to accept larger grants of existing software products. These licenses help us achieve our goal of providing reliable and long-lived software products through collaborative open source software development. In all cases, contributors retain full rights to use their original contributions for any other purpose outside of Apache while providing the ASF and its projects the right to distribute and build upon their work within Apache.

Apache License, Version 2.0 (current)

The 2.0 version of the Apache License was approved by the ASF in 2004. The goals of this license revision have been to reduce the number of frequently asked questions, to allow the license to be reusable without modification by any project (including non-ASF projects), to allow the license to be included by reference instead of listed in every file, to clarify the license on submission of contributions, to require a patent license on contributions that necessarily infringe the contributor's own patents, and to move comments regarding Apache and other inherited attribution notices to a location outside the license terms (the NOTICE file).

The result is a license that is supposed to be compatible with other open source licenses, while remaining true to the original goals of the Apache Group and supportive of collaborative development across both nonprofit and commercial organizations. The Apache Software Foundation is still trying to determine if this version of the Apache License is compatible with the GPL.
documentation on how to apply the Apache License to your work can be found in *<a href="../dev/apply-license.html">Applying the Apache License, Version 2.0</a>*.

<h3 id="1.1">Apache License, Version 1.1 (historic)</h3>

The 1.1 version of the Apache License was approved by the ASF in 2000. The primary change from the 1.0 license is in the 'advertising clause' (section 3 of the 1.0 license); derived products are no longer required to include attribution in their advertising materials, only in their documentation.

Individual packages licensed under the 1.1 version may have used different wording due to varying requirements for attribution or mark identification, but the binding terms were all the same.

<h3 id="1.0">Apache License, Version 1.0 (historic)</h3>

This is the original Apache License which applies only to older versions of Apache packages (such as version 1.2 of the Web server).

<h1 id="clas">Contributor License Agreements</h1>

The ASF desires that all contributors of ideas, code, or documentation to any Apache projects complete, sign, and submit (via fax or email) an *Individual Contributor License Agreement* (ICLA). The purpose of this agreement is to clearly define the terms under which intellectual property has been contributed to the ASF and thereby allow us to defend the project should there be a legal dispute regarding the software at some future time. A signed ICLA is required to be on file before an individual is given commit rights to an ASF project.

For a corporation that has assigned employees to work on an Apache project, a *Corporate CLA* (CCLA) is available for contributing intellectual property via the corporation, that may have been assigned as part of an employment agreement. Note that a Corporate CLA does not remove the need for every developer to sign their own ICLA as an individual, to cover any of their contributions which are not owned by the corporation signing the CCLA.

The ICLA is not tied to any employer you may have, so it is recommended to use one's personal email address in the contact details, rather than an @work address.

Your Full name will be published unless you provide an alternative Public name. For example if your full name is Andrew Bernard Charles Dickens, but you wish to be known as Andrew Dickens, please enter the latter as your Public name.

The email address and other contact details are not published.

<h1 id="grants">Software Grants</h1>

When an individual or corporation decides to donate a body of existing software or documentation to one of the Apache projects, they need to execute a formal *Software Grant Agreement* (SGA) with the ASF. Typically, this is done after negotiating approval with the ASF.
will not accept software unless there is a viable community available to support a collaborative project.

Documents may be submitted by fax or email.

If submitting by fax, please print, sign, and fax all pages of the document to +1-919-573-9199. Please send documents right-side-up, first page first; and send only one document per fax.

If submitting by email, please fill the form with a pdf viewer, print, sign, scan all pages into a single pdf file, and email the pdf file as an attachment to secretary@apache.org. If possible, send the attachment from the email address in the document. Please send only one document per email.

If you prefer to sign electronically, please fill the form, save it locally (e.g. icla.pdf), and sign the file by preparing a detached PGP signature. For example,

```
gpg --armor --detach-sign icla.pdf
```

The above will create a file icla.pdf.asc. Send both the file and signature as attachments in the same email to secretary@apache.org. Please send only one document (file plus signature) per email. Please do not submit your public key to Apache. Instead, please upload your public key to pgpkeys.mit.edu.

The files typically are named icla.pdf and icla.pdf.asc for individual agreements; ccla.pdf and ccla.pdf.asc for corporate agreements; software-grant.pdf and software-grant.pdf.asc for grants.

For export restriction information, please consult our ASF Export Classifications page.

For ASF trademark and logo usage information, please consult our ASF Trademark Use Policy page.

For answers to frequently asked licensing questions, please consult our Licensing Frequently Asked Questions page.

FAQs
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.
For additional credits (generally to people who reported problems) see CREDITS file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

---

Open Source Used In DNAC Platform Arica 11097
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
1.2335 jackson-jaxrs-base 2.8.4
1.2335.1 Available under license:
This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.2336.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.2337 twill-discovery-api 0.6.0-incubating

1.2337.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137747599_1614121902.52/0/twill-discovery-api-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/DiscoveryServiceClient.java
* /opt/cola/permits/1137747599_1614121902.52/0/twill-discovery-api-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/ServiceDiscovered.java
* /opt/cola/permits/1137747599_1614121902.52/0/twill-discovery-api-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/DiscoveryService.java
* /opt/cola/permits/1137747599_1614121902.52/0/twill-discovery-api-0-6-0-incubating-sources-2-
jar/org/apache/twill/discovery/Discoverable.java

1.2338 drill-storage-kafka 1.19.0

1.2338.1 Available under license:

Drill : Contrib : Storage : Kafka
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall meanLicensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing. Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
1.2339 map-visit 1.0.0 1.0.0
1.2339.1 Available under license:
The MIT License (MIT)
Copyright (c) 2015-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2340 kudu-client 1.14.0
1.2340.1 Available under license:
Apache Kudu
Copyright 2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were developed at
Open Source Used In DNAC Platform Arica 11106

Cloudera, Inc (http://www.cloudera.com/).

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

ggutil (some portions): Apache 2.0, and 3-clause BSD

Some portions of this module are derived from code in the Chromium project, copyright (c) Google inc and (c) The Chromium Authors and licensed under the Apache 2.0 License or the under the 3-clause BSD license:

Copyright (c) 2013 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------

src/kudu/gutil/utf: licensed under the following terms:

UTF-8 Library

The authors of this software are Rob Pike and Ken Thompson.

Copyright (c) 1998-2002 by Lucent Technologies.

Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software.

THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHORS NOR LUCENT TECHNOLOGIES MAKE ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

--------------------------------------------------------------------------------

src/kudu/util (some portions): 3-clause BSD license

Some portions of this module are derived from code from LevelDB
( https://github.com/google/leveldb ):

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------

src/kudu/util/array_view.h: 3-clause BSD license with patent grant

Copyright (c) 2015, The WebRTC project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Additional IP Rights Grant (Patents)

------------------------------------
"This implementation" means the copyrightable works distributed by Google as part of the WebRTC code package.

Google hereby grants to you a perpetual, worldwide, non-exclusive, no-charge, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, transfer, and otherwise run, modify and propagate the contents of this implementation of the WebRTC code package, where such license applies only to those patent claims, both currently owned by Google and acquired in the future, licensable by Google that are necessarily infringed by this implementation of the WebRTC code package. This grant does not include claims that would be infringed only as a consequence of further modification of this implementation. If you or your agent or exclusive licensee institute or order or agree to the institution of patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that this implementation of the WebRTC code package or any code incorporated within this implementation of the WebRTC code package constitutes direct or contributory patent infringement, or inducement of patent infringement, then any patent rights granted to you under this License for this implementation of the WebRTC code package shall terminate as of the date such litigation is filed.

--------------------------------------------------------------------------------

src/kudu/common/zp7.cc: MIT license

ZP7 (Zach's Peppy Parallel-Prefix-Popcountin' PEXT/PDEP Polyfill)

Copyright (c) 2020 Zach Wegner

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

--------------------------------------------------------------------------------

www/d3.v2.js: BSD 3-clause license

Copyright (c) 2012, Michael Bostock
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name Michael Bostock may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL MICHAEL BOSTOCK BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

--------------------------------------------------------------------------------

www/epoch.*: MIT license

Copyright (c) 2014 Fastly, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

--------------------------------------------------------------------------------

www/jquery*.js: MIT license


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

--------------------------------------------------------------------------------

java/kudu-client/: Some classes are derived from the AsyncHBase project under the following 3-clause BSD license:

Copyright (C) 2010-2012 The Async HBase Authors. All rights reserved. This file is part of Async HBase.

Redistribution and use in source and binary forms, with or without
Open Source Used In DNAC Platform Arica 11116

Modification, are permitted provided that the following conditions are met:
- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
- Neither the name of the StumbleUpon nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

src/kudu/util/x509_check_host.*: OpenSSL software license:

LICENSE ISSUES
=============

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of
the OpenSSL License and the original SSLeay license apply to the toolkit.
See below for the actual license texts.

OpenSSL License
===============

Copyright (c) 1998-2016 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License
---------------------

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).
The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as
the following conditions are heared to. The following conditions
apply to all code found in this distribution, be it the RC4, RSA,
lhash, DES, etc., code; not just the SSL code. The SSL documentation
included with this distribution is covered by the same copyright terms
except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in
the code are not to be removed.
If this package is used in a product, Eric Young should be given attribution
as the author of the parts of the library used.
This can be in the form of a textual message at program startup or
in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must display the following acknowledgement:
   "This product includes cryptographic software written by
   Eric Young (eay@cryptsoft.com)"
   The word 'cryptographic' can be left out if the routines from the library
   being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from
   the apps directory (application code) you must include an acknowledgement:
   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The licence and distribution terms for any publically available version or
derivative of this code cannot be changed. i.e. this code cannot simply be
The following dependencies or pieces of incorporated source code have licenses under one of the following categories:

(a) do not require their license text to be re-distributed with binary distributions (e.g. Boost license)

(b) have no requirements about re-distributing their license text in either source or binary distributions (e.g. public domain)

(c) require inclusion of the license text in both source and binary distributions, but are used only at build-time and thus not incorporated in binary artifacts (e.g. BSD- or MIT-licensed build tooling)

(d) are the same Apache 2.0 license reproduced in its entirety above.

Therefore, we do not reproduce their licenses in their entirety in this file.

 ycma_extra_conf.py: public domain
 cmake_modules/FindGMock.cmake: MIT license, build-only
 cmake_modules/FindProtobuf.cmake: BSD 3-clause license, build-only
 src/kudu/util (HdrHistogram-related classes): public domain
 src/kudu/gutil/valgrind.h: Hybrid BSD (half BSD, half zlib)
   - See the file headers for full license text.
 build-support/iwyu (some portions): LLVM Release License.
   - See thirdparty/LICENSE.txt for full license text.

 src/kudu/server/url-coding.cc (some portions): Boost software license

 www/bootstrap: Apache 2.0 license
 java/kudu-client/src/main/java/org/apache/kudu/util/{Slice,Slices}.java: Apache 2.0 license
 java/gradlew: Apache 2.0 license
   - See above for full text.

 Note on thirdparty dependencies downloaded during the build process

 The Kudu build process downloads many thirdparty dependencies automatically.
 Many of these dependencies are statically linked into the resulting Kudu binaries. Please refer to thirdparty/LICENSE.txt for relevant information.
1.2341 hadoop-hdfs-client 3.2.2

1.2341.1 Available under license:

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of org.iq80.leveledb:leveldb-api (https://github.com/dain/leveldb), which has the following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java), which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of Gson 2.2.4,
which has the following notices:

The Netty Project
=================
Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

-------------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* **HOMEPAGE:**
  * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* **LICENSE:**
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
* **HOMEPAGE:**
  * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* **LICENSE:**
  * license/LICENSE.jbzip2.txt (MIT License)
* **HOMEPAGE:**
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* **LICENSE:**
  * license/LICENSE.libdivsufsort.txt (MIT License)
* **HOMEPAGE:**
  * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* **LICENSE:**
  * license/LICENSE.jctools.txt (ASL2 License)
* **HOMEPAGE:**
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* **LICENSE:**
  * license/LICENSE.jzlib.txt (BSD style License)
* **HOMEPAGE:**
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:
This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced
by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance
non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.alto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.
Original source copyright:Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of
Apache Log4j 1.2.17,
which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative
  Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
Official home: http://www.jcip.net Any republication or derived work
distributed in source code form must include this copyright and license
notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
  org.apache.tomcat:tomcat-jasper
  org.apache.tomcat:tomcat-juli
  org.apache.tomcat:tomcat-jsp-api
  org.apache.tomcat:tomcat-el-api
  org.apache.tomcat:tomcat-jasper-el
  org.apache.tomcat:tomcat-api
  org.apache.tomcat:tomcat-util-scan
  org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
  org.apache.tomcat:tomcat-jasper-el
  org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jettyschemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt
for non-commercial or commercial purposes and without fee is
granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of
Snappy for Java 1.0.4.1,
which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by
"GCC RUnetime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of
Xerces2 Java Parser 2.9.1,
which has the following notices:
* =========================================================================
  ==  NOTICE file corresponding to section 4(d) of the Apache License,   ==
  ==  Version 2.0, in this case for the Apache Xerces Java distribution.  ==
  =========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

  This product includes software developed at
  The Apache Software Foundation (http://www.apache.org/).

  --
  This product incorporates portions of the 'Hadoop' project

  Copyright 2007-2009 The Apache Software Foundation

  Licensed under the Apache License v2.0

  --
  Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
  It is licensed Creative Commons Attribution 3.0.
  See https://creativecommons.org/licenses/by/3.0/us/
  We changed the logo by stripping the colored background, inverting
  it and then rotating it some.

  Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0.
The license is bundled with the download available here:
--
This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.
http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--
This product includes portions of the Guava project v14, specifically
'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of
Phoenix Core 4.7.0,
which has the following notices:
  Apache Phoenix
  Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library
distributed under the terms of the Apache 2 license. Original source copyright:
  Copyright 2014 Simply Measured, Inc.
  Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5
(https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =

This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/\*\*
 *\n * Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)\n * All rights reserved.\n * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:\n * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.\n * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.\n * Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.\n *\n * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.\n */
For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/

/*++
 *
* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
* 
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html

* Abstract: The main routine
* 
--*/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/\{lz4.h,lz4.c,lz4hc.h,lz4hc.c\}.
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

*/
copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:
re2j 1.1

----------------------------------------------------------------------
(GO license)
This is a work derived from Russ Cox’s RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h
-------------------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni
(https://github.com/fusesource/leveldbjni), which is available under the
following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy (http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

-----------------------------------------------------------------------------------------------------------------------------
Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

-----------------------------------------------------------------------------------------------------------------------------
Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

--------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the
following license:

- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
- hadoop-tools/hadoop-sls/src/main/js/thirdparty/bootstrap.min.js
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
- hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
- bootstrap v3.3.6
- broccoli-asset-rev v2.4.2
- broccoli-funnel v1.0.1
- datatables v1.10.8
- em-helpers v0.5.13
- em-table v0.1.6
- ember v2.2.0
- ember-array-contains-helper v1.0.2
- ember-bootstrap v0.5.1
- ember-cli v1.13.13
- ember-cli-app-version v1.0.0
- ember-cli-babel v5.1.6
- ember-cli-content-security-policy v0.4.0
- ember-cli-dependency-checker v1.2.0
- ember-cli-htmlbars v1.0.2
- ember-cli-htmlbars-inline-precompile v0.3.1
- ember-cli-ic-ajax v0.2.1
- ember-cli-inject-live-reload v1.4.0
- ember-cli-jquery-ui v0.0.20
- ember-cli-qunit v1.2.1
- ember-cli-release v0.2.8
- ember-cli-shims v0.0.6
- ember-cli-sri v1.2.1
- ember-cli-test-loader v0.2.1
- ember-cli-uglify v1.2.0
- ember-d3 v0.1.0
- ember-data v2.1.0
- ember-disable-proxy-controllers v1.0.1
- ember-export-application-global v1.0.5
- ember-load-initializers v0.1.7
- ember-qunit v0.4.16
- ember-qunit-notifications v0.1.0
- ember-resolver v2.0.3
- ember-spin-spinner v0.2.3
- ember-truth helpers v1.2.0
- jquery v2.1.4
- jquery-ui v1.11.4
- loader.js v3.3.0
- momentjs v2.10.6
- qunit v1.19.0
- select2 v4.0.0
- snippet-ss v1.11.0
- spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodeings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:

hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the
following license:
- HSQLDB Database 2.3.4

(HSQL License)

"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; 
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND 
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS 
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group. 
All rights reserved. 
Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this 
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, 
this list of conditions and the following disclaimer in the documentation 
and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its 
contributors may be used to endorse or promote products derived from this 
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS""
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, 
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; 
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND 
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS 
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the 
Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of
computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with
Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices. You must
include a notice in each of Your Modifications that identifies You as the
Contributor of the Modification. You may not remove or alter any copyright,
patent or trademark notices contained within the Covered Software, or any
notices of licensing or any descriptive text giving attribution to any
Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code
form that alters or restricts the applicable version of this License or the
recipients rights hereunder. You may choose to offer, and to charge a fee for,
warrranty, support, indemnity or liability obligations to one or more recipients
of Covered Software. However, you may do so only on Your own behalf, and not on
behalf of the Initial Developer or any Contributor. You must make it absolutely
clear that any such warranty, support, indemnity or liability obligation is
offered by You alone, and You hereby agree to indemnify the Initial Developer
and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of warranty, support, indemnity or liability terms
You offer.

3.5. Distribution of Executable Versions. You may distribute the
Executable form of the Covered Software under the terms of this License or under
the terms of a license of Your choice, which may contain terms different from
this License, provided that You are in compliance with the terms of this License
and that the license for the Executable form does not attempt to limit or alter
the recipients rights in the Source Code form from the rights set forth in this
License. If You distribute the Covered Software in Executable form under a
different license, You must make it absolutely clear that any terms which differ
from this License are offered by You alone, not by the Initial Developer or
Contributor. You hereby agree to indemnify the Initial Developer and every
Contributor for any liability incurred by the Initial Developer or such
Contributor as a result of any such terms You offer.

3.6. Larger Works. You
may create a Larger Work by combining Covered Software with other code not
governed by the terms of this License and distribute the Larger Work as a single
product. In such a case, You must make sure the requirements of this License are
fulfilled for the Covered Software.

4. Versions of the License.

4.1.
New Versions. Sun Microsystems, Inc. is the initial license steward and may
publish revised and/or new versions of this License from time to time. Each
version will be given a distinguishing version number. Except as provided in
Section 4.3, no one other than the license steward has the right to modify this

Open Source Used In DNAC Platform Arica 11156
4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial
Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the
jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or
contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants
You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual
property rights (other than patent or trademark) Licensable by Initial
Developer, to use, reproduce, modify, display, perform, sublicense and
distribute the Original Software (or portions thereof), with or without
Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims
infringed by the making, using or selling of Original Software, to make, have
made, use, practice, sell, and offer for sale, and/or otherwise dispose of the
Original Software (or portions thereof).
(c) The licenses granted in Sections
2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms
of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or (2) for
infringements caused by: (i) the modification of the Original Software, or (ii)
the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and
subject to third party intellectual property claims, each Contributor hereby
grants You a world-wide, royalty-free, non-exclusive license:
(a) under
intellectual property rights (other than patent or trademark) Licensable by
Contributor to use, reproduce, modify, display, perform, sublicense and
distribute the Modifications created by such Contributor (or portions thereof),
either on an unmodified basis, with other Modifications, as Covered Software
and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such
combination), to make, use, sell, offer for sale, have made, and/or otherwise
dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such
Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND
LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered
Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0
This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT’S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and

b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

FindBugs-jsr305 3.0.0

dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:

"Java Concurrency in Practice" book annotations 1.0

(CCAL v2.5)

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as
incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission Derivative Works.
For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive
right to collect, whether individually or via a performance rights society (e.g.
ASCAP, BMI, SESAC), royalties for the public performance or public digital
performance (e.g. webcast) of the Work.
Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right
to collect, whether individually or via a music rights agency or designated
agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created
by 17 USC Section 115 of the US Copyright Act (or the equivalent in other
jurisdictions).
Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the
Work is a sound recording, Licensor waives the exclusive right to collect,
whether individually or via a performance-rights society (e.g. SoundExchange),
royalties for the public digital performance (e.g. webcast) of the Work, subject
to the compulsory license created by 17 USC Section 114 of the US Copyright Act
(or the equivalent in other jurisdictions).
The above rights may be exercised in all media and formats whether now known or
hereafter devised. The above rights include the right to make such modifications
as are technically necessary to exercise the rights in other media and formats.
All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject
to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally
perform the Work only under the terms of this License, and You must include a
copy of, or the Uniform Resource Identifier for, this License with every copy or
phonorecord of the Work You distribute, publicly display, publicly perform, or
publicly digitally perform. You may not offer or impose any terms on the Work
that alter or restrict the terms of this License or the recipients' exercise of
the rights granted hereunder. You may not sublicense the Work. You must keep
intact all notices that refer to this License and to the disclaimer of
warranties. You may not distribute, publicly display, publicly perform, or
publicly digitally perform the Work with any technological measures that control
access or use of the Work in a manner inconsistent with the terms of this
License Agreement. The above applies to the Work as incorporated in a Collective
Work, but this does not require the Collective Work apart from the Work itself
to be made subject to the terms of this License. If You create a Collective
Work, upon notice from any Licensor You must, to the extent practicable, remove
from the Collective Work any credit as required by clause 4(b), as requested. If
You create a Derivative Work, upon notice from any Licensor You must, to the
extent practicable, remove from the Derivative Work any credit as required by
clause 4(b), as requested.
If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, a credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any
such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

--------------------------------------------------------------------------------
(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributors Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.
1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.
1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a
timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent
prohibited by statute or regulation, such description must be sufficiently
detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You
fail to comply with any of its terms. However, if You become compliant, then the
rights granted under this License from a particular Contributor are reinstated
(a) provisionally, unless and until such Contributor explicitly and finally
terminates Your grants, and (b) on an ongoing basis, if such Contributor fails
to notify You of the non-compliance by some reasonable means prior to 60 days
after You have come back into compliance. Moreover, Your grants from a
particular Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the first
time You have received notice of non-compliance with this License from such
Contributor, and You become compliant prior to 30 days after Your receipt of the
notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions, counter-claims, and
cross-claims) alleging that a Contributor Version directly or indirectly
infringes any patent, then the rights granted to You by any and all Contributors
for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user
license agreements (excluding distributors and resellers) which have been
validly granted by You or Your distributors under this License prior to
termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without
warranty of any kind, either expressed, implied, or statutory, including,
without limitation, warranties that the Covered Software is free of defects,
merchantable, fit for a particular purpose or non-infringing. The entire risk as
to the quality and performance of the Covered Software is with You. Should any
Covered Software prove defective in any respect, You (not any Contributor)
assume the cost of any necessary servicing, repair, or correction. This
disclaimer of warranty constitutes an essential part of this License. No use of
any Covered Software is authorized under this License except under this
disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including
negligence), contract, or otherwise, shall any Contributor, or anyone who
distributes Covered Software as permitted above, be liable to You for any
direct, indirect, special, incidental, or consequential damages of any character
including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).
10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

="/--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.
4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED "$ AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:

Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work,
the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.
2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish." or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:
a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create a Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this
License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.
Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2342 jersey-common 2.25.1

1.2342.1 Available under license:

  CDDL+GPL License
  http://glassfish.java.net/public/CDDL+GPL_1_1.html

1.2343 jackson-jaxrs 2.10.4

1.2343.1 Available under license:

  This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").
  See the License for details about distribution rights, and the specific rights regarding derivates works.

  You may obtain a copy of the License at:

  http://www.apache.org/licenses/LICENSE-2.0
  # Jackson JSON processor

  Jackson is a high-performance, Free/Open Source JSON processing library.
  It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
  It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

  ## Licensing

  Jackson core and extension components may be licensed under different licenses.
  To find the details that apply to this artifact see the accompanying LICENSE file.
  For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

  ## Credits

  A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.2344 caniuse-lite 1.0.30001202

1.2344.1 Available under license:

  Attribution 4.0 International

=======================================================================
Creative Commons Corporation (“Creative Commons”) is not a law firm and does not provide legal services or legal advice. Distribution of Creative Commons public licenses does not create a lawyer-client or other relationship. Creative Commons makes its licenses and related information available on an “as-is” basis. Creative Commons gives no warranties regarding its licenses, any material licensed under their terms and conditions, or any related information. Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible.

Using Creative Commons Public Licenses

Creative Commons public licenses provide a standard set of terms and conditions that creators and other rights holders may use to share original works of authorship and other material subject to copyright and certain other rights specified in the public license below. The following considerations are for informational purposes only, are not exhaustive, and do not form part of our licenses.

Considerations for licensors: Our public licenses are intended for use by those authorized to give the public permission to use material in ways otherwise restricted by copyright and certain other rights. Our licenses are irrevocable. Licensors should read and understand the terms and conditions of the license they choose before applying it. Licensors should also secure all rights necessary before applying our licenses so that the public can reuse the material as expected. Licensors should clearly mark any material not subject to the license. This includes other CC-licensed material, or material used under an exception or limitation to copyright. More considerations for licensors: wiki.creativecommons.org/Considerations_for_licensors

Considerations for the public: By using one of our public licenses, a licensor grants the public permission to use the licensed material under specified terms and conditions. If the licensor's permission is not necessary for any reason—for example, because of any applicable exception or limitation to copyright—then that use is not regulated by the license. Our licenses grant only permissions under copyright and certain other rights that a licensor has authority to grant. Use of the licensed material may still be restricted for other reasons, including because others have copyright or other rights in the material. A licensor may make special requests, such as asking that all changes be marked or described. Although not required by our licenses, you are encouraged to respect those requests where reasonable. More considerations for the public:
Creative Commons Attribution 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 -- Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.

b. Adapter's License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights
that applies to Your use of the Licensed Material.

f. Licensed Material means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

g. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. Licensor means the individual(s) or entity(ies) granting rights under this Public License.

i. Share means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

k. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

Section 2 -- Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

   a. reproduce and Share the Licensed Material, in whole or in part; and

   b. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public
3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, simply making modifications authorized by this Section 2(a)(4) never produces Adapted Material.

5. Downstream recipients.

a. Offer from the Licensor -- Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

b. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed
Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.

3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 -- License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

1. If You Share the Licensed Material (including in modified form), You must:

   a. retain the following if it is supplied by the Licensor with the Licensed Material:

      i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);

      ii. a copyright notice;

      iii. a notice that refers to this Public License;

      iv. a notice that refers to the disclaimer of warranties;

      v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;

   b. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and

   c. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.
2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or hyperlink to a resource that includes the required information.

3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.

4. If You Share Adapted Material You produce, the Adapter's License You apply must not prevent recipients of the Adapted Material from complying with this Public License.

Section 4 -- Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;

b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and

c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 -- Disclaimer of Warranties and Limitation of Liability.

a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS,
ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE, WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.

b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.

c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 -- Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:

1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

2. upon express reinstatement by the Licensor.

For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.
Section 7 -- Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.

b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 -- Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.

d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

=======================================================================

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the Licensor. The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including.
without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

Creative Commons may be contacted at creativecommons.org.

1.2345 jline 2.9

1.2346 netty 4.1.61.Final

1.2346.1 Available under license:

Copyright (c) 2003-2008 Yuta Mori All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this
work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2009 William Kinney

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright 2009-2010 Ning, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*
 * Copyright (c) 2004-2006, 2008, 2009, 2011 Apple Inc. All rights reserved.
 *
 * @APPLE_LICENSE_HEADER_START@
 *
 * This file contains Original Code and/or Modifications of Original Code as defined in and that are subject to the Apple Public Source License Version 2.0 (the 'License'). You may not use this file except in compliance with the License. Please obtain a copy of the License at https://www.opensource.apple.com/apsl/ and read it before using this file.
 *
 * The Original Code and all software distributed under the License are distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT.
 * Please see the License for the specific language governing rights and
The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2010-2011 Matthew J. Francis and Contributors of the jbzip2 Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuate of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial,
and in any way, including by methods that have not yet been invented or conceived.

The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Netty Project
=================

Please visit the Netty web site for more information:

* https://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

-----------------------------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * https://www.slf4j.org/

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* NOTICE:
  * license/NOTICE.harmony.txt
* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
* HOMEPAGE:
  * https://archive.apache.org/dist/harmony/

This product contains a modified portion of 'jbzip2', a Java bzip2 compression
and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
* HOMEPAGE:
  * https://github.com/y-256/libdivsufsort

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
* HOMEPAGE:
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java
This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/protobuf

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * https://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/snappy

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jboss-remoting/jboss-marshalling
This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/caliper

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * https://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Cory Benfield. It can be obtained at:

* LICENSE:
  * license/LICENSE.hyper-hpack.txt (MIT License)
* HOMEPAGE:
  * https://github.com/python-hyper/hpack/
This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Tatsuhiro Tsujikawa. It can be obtained at:

* LICENSE:  
  * license/LICENSE.nghttp2-hpack.txt (MIT License)  
* HOMEPAGE:  
  * https://github.com/nghttp2/nghttp2/

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:  
  * license/LICENSE.commons-lang.txt (Apache License 2.0)  
* HOMEPAGE:  
  * https://commons.apache.org/proper/commons-lang/

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:  
  * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)  
* HOMEPAGE:  
  * https://github.com/takari/maven-wrapperm

This product contains the dnsinfo.h header file, that provides a way to retrieve the system DNS configuration on MacOS. This private header is also used by Apple's open source mDNSResponder (https://opensource.apple.com/tarballs/mDNSResponder/).

* LICENSE:  
  * license/LICENSE.dnsinfo.txt (Apple Public Source License 2.0)  
* HOMEPAGE:  
  Apache License  
  Version 2.0, January 2004  
  https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

(BSD License: https://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Webbit nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

/*
 * Copyright (c) 2004-2007 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Protocol Buffers - Google's data interchange format
Copyright 2013 Google Inc. All rights reserved.
https://developers.google.com/protocol-buffers/

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
This copy of Aalto XML processor is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:
https://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"

Apache Harmony

Copyright 2006, 2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

1.2347 esri-geometry-api 2.2.0

1.2347.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
   Copyright 1995-2017 Esri

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at
   http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/JSONUtils.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfInt8.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/ogc/OGCCurve.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/JsonGeometryException.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OGCMultiSurface.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OGCMultiPoint.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/JsonReader.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/JsonParserReader.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorFactoryLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OGCMultiCurve.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorCentroid2DLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OGCLinearRing.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfInt64.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfInt16.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfFloat.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfInt32.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/SimpleJsonReaderCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromGeoJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJsonCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromGeoJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromJson.java
* /opt(ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/ OperatorCentroid2DLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorCentroid2D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorImportFromGeoJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Operator.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OGCLinearRing.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfFloat.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/AttributeStreamOfDbl.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Point3D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/envelope3d-Java-gz/Envelope3D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/RasterizedGeometry2Dimpl.java

No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2017 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/
/*
COPYRIGHT 1995-2017 ESRI

TRADE SECRETS: ESRI PROPRIETARY AND CONFIDENTIAL
Unpublished material - all rights reserved under the
Copyright Laws of the United States.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/JsonReaderCursor.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 2013-2015 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Open Source Used In DNAC Platform Arica 11234

Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/CombineOperator.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/CombineOperator.java
No license file was found, but licenses were detected in source scan.

/*
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Open Source Used In DNAC Platform Arica 11234
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorExportToGeoJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/OperatorExportToGeoJsonLocal.java

No license file was found, but licenses were detected in source scan.

*/

Copyright 1995-2015 Esri

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorShapePreservingDensifyLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorExportToJson.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/VertexDescriptionHash.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SpatialReferenceImpl.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MathUtils.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Envelope3D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorSymmetricDifferenceLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDistance.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Treap.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorUnionLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorRelateLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorWithinLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDensifyByLengthCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/InternalUtils.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorProximity2DLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorIntersectsLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/WkbExportFlags.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Proximity2DResultComparator.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/BucketSort.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MultiVertexGeometryImpl.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorSimplifyLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/GenericGeometrySerializer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Envelope1D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MapOGCStructure.java
Open Source Used In DNAC Platform Arica 11238

jar/com/esri/core/geometry/RelationalOperationsMatrix.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/VertexDescription.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorGeodeticAreaLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ByteBufferCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorConvexHullLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorOverlapsLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDistanceLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorImportFromWkb.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorClip.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/RingOrientationFixer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/PtSrlzr.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/RasterizedGeometry2D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorExportToESRIShapeCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorEquals.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/TopoGraph.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorUnionCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SegmentIteratorImpl.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/WktExportFlags.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/JsonStringWriter.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/AttributeStreamBase.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorIntersectionCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorConvexHullCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ECoordinate.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
Open Source Used In DNAC Platform Arica 11240
jar/com/esri/core/geometry/OperatorExportToJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorCutCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/UserCancelException.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/IndexMultiDCList.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/EnvSrlzr.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorExportToJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SweepComparator.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Clusterer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SegmentIntersector.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDifference.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorUnion.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Boundary.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ClassicSort.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorRelate.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Point.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorBufferLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ProjectionTransformation.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/GeoDist.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Clusterer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SegmentIntersector.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDifference.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Boundary.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ClassicSort.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorRelate.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorWithin.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorCrossesLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Cutter.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorSimplify.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MultiPathImpl.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/WkbGeometryType.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorProximity2D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/GeometryException.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorGeodeticDensityByLength.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorOverlaps.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/JsonWriter.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ShapeModifiers.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorOffsetLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorGeneralize.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/NumberUtils.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SegmentBuffer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorClipCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/QuadTree.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MapGeometryCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDisjointLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorImportFromWkbLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/ConstructOffset.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorTouchesLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorImportFromESRIShapeLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorEqualsLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorDifferenceCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/WkbByteOrder.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorCut.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/MultiPoint.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorProjectLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorExportToWktLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorExportToWkt.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorOffsetCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/WktParser.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/Simplificator.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/DirtyFlags.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/StringUtils.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorImportFromWktLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorSimplifyCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorSymmetricDifference.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/SimpleByteBufferCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/IndexHashTable.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/GeometrySerializer.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorBoundaryLocalCursor.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/OperatorIntersectionLocal.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-
jar/com/esri/core/geometry/QuadTreeImpl.java

No license file was found, but licenses were detected in source scan.

/*

Copyright 1995-2018 Esri
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com

*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/ogc/OGCGeometry.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Polyline.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Geometry.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/MapGeometry.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/SizeOf.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Point2D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Envelope2D.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/ogc/OGCPolygon.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/SpatialReference.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/GeometryEngine.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Polygon.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/Envelope.java
* /opt/ws_local/PERMITS_SQL/1066571557_1593680380.14/0/esri-geometry-api-2-2-0-sources-jar/com/esri/core/geometry/ogc/OGCLineString.java
Open Source Used In DNAC Platform Arica

1.2348 engine.io-parser 2.2.1 2.2.1

1.2348.1 Available under license:

(The MIT License)

Copyright (c) 2016 Guillermo Rauch (@rauchg)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2349 disruptor 3.4.2

1.2349.1 Available under license:

No license file was found, but licenses were detected in source scan.

/* Copyright 2016 Gil Tene
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-
jar/com/lmax/disruptor/util/ThreadHints.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 LMAX Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-
jar/com/lmax/disruptor/EventReleaser.java
*/opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-
jar/com/lmax/disruptor/EventReleaseAware.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 LMAX Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AggregateEventHandler.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/EventFactory.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/SequenceReportingEventHandler.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/SleepingWaitStrategy.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/ExceptionHandler.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/EventProcessor.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/SequenceBarrier.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/BusySpinWaitStrategy.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/EventTranslatorTwoArg.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/IgnoreExceptionHandler.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/NoOpEventProcessor.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/BatchEventProcessor.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/dsl/ExceptionHandlerSetting.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/WorkerPool.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/util/DaemonThreadFactory.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/LiteBlockingWaitStrategy.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/dsl/EventProcessorInfo.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/PhasedBackoffWaitStrategy.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/SingleProducerSequencer.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AbstractSequencer.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AbstractEventProcessor.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AbstractEventTranslator.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/EventTranslator.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AbstractWaitStrategy.java
- /opt/cola/permits/1005594541_1646171709.07/0/disruptor-3-4-2-sources-jar/com/lmax/disruptor/AbstractEventFactory.java
/*
 * Copyright 2012 LMAX Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */

No license file was found, but licenses were detected in source scan.


1.2350 jackson-databind 2.10.5.1

1.2350.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

### 1.2351 punycode 1.4.1 1.4.1

#### 1.2351.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.2352 util-deprecate 1.0.2 1.0.2

#### 1.2352.1 Available under license:

(The MIT License)

Copyright (c) 2014 Nathan Rajlich <nathan@tootallnate.net>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2353 arraybuffer.slice 0.0.6 0.0.6
1.2353.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2354 deep-eql 3.0.1
1.2354.1 Available under license:

Copyright (c) 2013 Jake Luer <jake@alogicalparadox.com> (http://alogicalparadox.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2355 react-base16-styling 0.6.0 0.6.0

1.2355.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Alexander Kuznetsov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2356 thrift 0.9.3

1.2356.1 Available under license:

This package was debianized by Thrift Developer's <dev@thrift.apache.org>.

This package and the Debian packaging is licensed under the Apache License, see `/usr/share/common-licenses/Apache-2.0`.

Open Source Used In DNAC Platform Arica 11253
The following informations was copied from Apache Thrift LICENSE file.

--------------------------------------------------
SOFTWARE DISTRIBUTED WITH THRIFT:
--------------------------------------------------

The Apache Thrift software includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

--------------------------------------------------

Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

--------------------------------------------------

The following files contain some portions of code contributed under the Thrift Software License (see doc/old-thrift-license.txt), and relicensed under the Apache 2.0 License:

compiler/cpp/Makefile.am
compiler/cpp/src/generate/t_cocoa_generator.cc
compiler/cpp/src/generate/t_cpp_generator.cc
compiler/cpp/src/generate/t_csharp_generator.cc
compiler/cpp/src/generate/t_erl_generator.cc
compiler/cpp/src/generate/t.hs_generator.cc
compiler/cpp/src/generate/t_java_generator.cc
compiler/cpp/src/generate/t_ocaml_generator.cc
compiler/cpp/src/generate/t_perl_generator.cc
compiler/cpp/src/generate/t_php_generator.cc
compiler/cpp/src/generate/t.py_generator.cc
compiler/cpp/src/generate/t.rb_generator.cc
compiler/cpp/src/generate/t_st_generator.cc
compiler/cpp/src/generate/t.xsd_generator.cc
compiler/cpp/src/main.cc
compiler/cpp/src/parse/t_field.h
compiler/cpp/src/parse/t_program.h
compiler/cpp/src/platform.h
compiler/cpp/src/thriftl.ll
compiler/cpp/src/thrifty.yy
lib/csharp/src/Protocol/TBinaryProtocol.cs
lib/csharp/src/Protocol/TField.cs
lib/csharp/src/Protocol/TList.cs
lib/csharp/src/Protocol/TMap.cs
For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

#  Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#  
#  Copying and distribution of this file, with or without
#  modification, are permitted in any medium without royalty provided
#  the copyright notice and this notice are preserved.

For the compiler/cpp/src/md5.[ch] components:

/*

Copyright (C) 1999, 2000, 2002 Aladdin Enterprises. All rights reserved.
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

L. Peter Deutsch
ghost@aladdin.com

*/

------------------------------------------------------------------------
For the lib/rb/setup.rb: Copyright (c) 2000-2005 Minero Aoki,
lib/ocaml/OCamlMakefile and lib/ocaml/README-OCamlMakefile components:
  Copyright (C) 1999 - 2007 Markus Mottl

Licensed under the terms of the GNU Lesser General Public License 2.1
(see doc/lgpl-2.1.txt for the full terms of this license)

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the 
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity 
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, 
including but not limited to software source code, documentation 
source, and configuration files.

"Object" form shall mean any form resulting from mechanical 
transformation or translation of a Source form, including but 
not limited to compiled object code, generated documentation, 
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or 
Object form, made available under the License, as indicated by a 
copyright notice that is included in or attached to the work 
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object 
form, that is based on (or derived from) the Work and for which the 
editorial revisions, annotations, elaborations, or other modifications 
represent, as a whole, an original work of authorship. For the purposes 
of this License, Derivative Works shall not include works that remain 
separable from, or merely link (or bind by name) to the interfaces of, 
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including 
the original version of the Work and any modifications or additions 
to that Work or Derivative Works thereof, that is intentionally 
submitted to Licensor for inclusion in the Work by the copyright owner 
or by an individual or Legal Entity authorized to submit on behalf of 
the copyright owner. For the purposes of this definition, "submitted" 
means any form of electronic, verbal, or written communication sent 
to the Licensor or its representatives, including but not limited to 
communication on electronic mailing lists, source code control systems, 
and issue tracking systems that are managed by, or on behalf of, the 
Licensor for the purpose of discussing and improving the Work, but 
excluding communication that is conspicuously marked or otherwise 
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity 
on behalf of whom a Contribution has been received by Licensor and 
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of 
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SOFTWARE DISTRIBUTED WITH THRIFT:

The Apache Thrift software includes a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.
Portions of the following files are licensed under the MIT License:

lib/erl/src/Makefile.am

Please see doc/otp-base-license.txt for the full terms of this license.

--------------------------------------------------
For the aclocal/ax_boost_base.m4 and contrib/fb303/aclocal/ax_boost_base.m4 components:

#   Copyright (c) 2007 Thomas Porschberg <thomas@randspringer.de>
#
#   Copying and distribution of this file, with or without
#   modification, are permitted in any medium without royalty provided
#   the copyright notice and this notice are preserved.

--------------------------------------------------
For the compiler/cpp/src/thrift/md5.[ch] components:

/*
   Copyright (C) 1999, 2000, 2002 Aladdin Enterprises. All rights reserved.

   This software is provided 'as-is', without any express or implied
   warranty. In no event will the authors be held liable for any damages
   arising from the use of this software.

   Permission is granted to anyone to use this software for any purpose,
   including commercial applications, and to alter it and redistribute it
   freely, subject to the following restrictions:

   1. The origin of this software must not be misrepresented; you must not
      claim that you wrote the original software. If you use this software
      in a product, an acknowledgment in the product documentation would be
      appreciated but is not required.
   2. Altered source versions must be plainly marked as such, and must not be
      misrepresented as being the original software.
   3. This notice may not be removed or altered from any source distribution.

   L. Peter Deutsch
   ghost@aladdin.com

   */

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

Open Source Used In DNAC Platform Arica 11261
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
Permission is hereby granted, free of charge, to any person obtaining a copy of this software (OTP Base, fslib, G.A.S) and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Thrift
Copyright 2006-2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2357 from 0.1.7 0.1.7

1.2357.1 Available under license:

The MIT License

Copyright (c) 2011 Dominic Tarr

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License, Version 2.0

Copyright (c) 2011 Dominic Tarr

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2358 string-prototype-trimend 1.0.4
1.2358.1 Available under license:
MIT License

Copyright (c) 2017 Khaled Al-Ansari

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2359 jetty-util 6.1.26

1.2360 annotations 4.1.1.4
1.2360.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Android Open Source Project
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
1.2361 commons-jexl 2.1.1

1.2361.1 Available under license:
Apache Commons JEXL
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2362 junit-platform-engine 1.7.0
1.2362.1 Available under license:
Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECEPIENT'S ACCEPTANCE OF THIS AGREEMENT.

### 1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
  * **i)** changes to the Program, and
  * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself
or anyone acting on such Contributor’s behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

### 2. Grant of Rights

**a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

**b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

**c)** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any
other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

**d)** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

**e)** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient’s receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

### 3. Requirements

**3.1** If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
  * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
  * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
  * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
  * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

**3.2** When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
  * **b)** a copy of this Agreement must be included with each copy of the Program.

**3.3** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

### 4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

### 5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

### 6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### 7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

#### Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}. 

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.
1.2363 isarray 2.0.1 2.0.1

1.2363.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2364 array-includes 3.1.4

1.2364.1 Available under license:

The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2365 unorm 1.6.0 1.6.0

1.2365.1 Available under license:

The software dual licensed under the MIT and GPL licenses. MIT license:

Copyright (c) 2008-2013 Matsuza <matsuza@gmail.com>, Bjarke Walling <bwp@bwp.dk>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GPL notice (please read the [full GPL license] online):

Copyright (C) 2008-2013 Matsuza <matsuza@gmail.com>, Bjarke Walling <bwp@bwp.dk>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

[full GPL license]: http://www.gnu.org/licenses/gpl-2.0-standalone.html

### 1.2366 figures 1.7.0

**1.2366.1 Available under license:**

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.2367 media-typer 0.3.0 0.3.0

**1.2367.1 Available under license:**

(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2368 @babel/plugin-transform-sticky-regex
7.16.0
1.2368.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2369 es6-weak-map 2.0.3
1.2369.1 Available under license :
ISC License

Copyright (c) 2013-2018, Mariusz Nowak, @medikoo, medikoo.com
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2370 arr-flatten 1.1.0 1.1.0

1.2370.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2371 yargs-parser 20.2.7 20.2.7

1.2371.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice
1.2372 ienoopen 1.0.0 1.0.0

1.2372.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014, 2015 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2373 duplexer3 0.1.4

1.2373.1 Available under license :
Copyright (c) 2013, Deoxxa Development

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of Deoxxa Development nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY DEOXXA DEVELOPMENT "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL DEOXXA DEVELOPMENT BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2374 zlib 1.2.5

1.2374.1 Available under license :
/* zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jlosp@gzip.org          madler@alumni.caltech.edu

*/
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2375 object-inspect 1.9.0
1.2375.1 Available under license :
MIT License

Copyright (c) 2013 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
1.2376 urix 0.1.0 0.1.0

1.2376.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2377 babel-plugin-transform-class-properties 6.24.1

1.2377.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2378 snake-yaml 1.12

1.2378.1 Available under license:

Found license 'GNU Lesser General Public License' in '// This module is multi-licensed and may be used under the
terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General
Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // LGPL, GNU General Public License, V2 or
BSD.'

Found license 'Eclipse Public License 1.0' in '// This module is multi-licensed and may be used under the terms //
EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public
1.2379 babel-helper-call-delegate 6.24.1 6.24.1

1.2379.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2380 @babel/plugin-transform-exponentiation-operator 7.16.0

1.2380.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingListMultimap.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Ascii.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/annotations/package-info.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingSetMultimap.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ListeningExecutorService.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ForwardingBlockingQueue.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/SortedMapDifference.java
  * /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/UnmodifiableListIterator.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingImmutableCollection.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/net/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/primitives/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/RowSortedTable.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Strings.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/UncaughtExceptionHandlers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/AbstractSequentialIterator.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ThreadFactoryBuilder.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/annotations/Beta.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/MinMaxPriorityQueue.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/RegularImmutableMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/ImmutableAsList.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/ImmutableSortedMapFauxverideShim.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/base/Platform.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/ImmutableEnumSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/Escapers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/DiscreteDomain.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/annotations/GwtIncompatible.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/AbstractExecutionThreadService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/RegularImmutableSortedSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/RegularImmutableList.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/ComparisonChain.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/AbstractIdleService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/EmptyImmutableSetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/ArrayTable.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/escape/Platform.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/SingletonImmutableTable.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/SingletonImmutableList.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/AbstractService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/ImmutableTable.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/ImmutableSetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/net/MediaType.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/net/MediaType.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ListeningScheduledExecutorService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/FutureCallback.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/FutureFallback.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Optional.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/math/LongMath.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/reflect/Types.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ForwardingCheckedFuture.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/AbstractCompositeHashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/math/BigIntegerMath.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Present.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Ticker.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/cache/ForwardingCache.java

Open Source Used In DNAC Platform Arica  11297
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2006 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/Futures.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/CharEscaper.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/WrappingScheduledExecutorService.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/Runnables.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/io/CharSequenceReader.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/SortedMultiset.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/eventbus/SubscriberExceptionContext.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/GwtWorkarounds.java

* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ImmutableRangeMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/Queues.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/AbstractSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/HashCode.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/AbstractHasher.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Funnels.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Hashing.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/MessageDigestHashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/BoundType.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/EmptyImmutableSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Murmur3_128HashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Funnel.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/hash/AbstractStreamingHashFunction.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/TreeRangeSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/hash/Hasher.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/hash/BloomFilterStrategies.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/hash/BloomFilter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/hash/PrimitiveSink.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/hash/HashingOutputStream.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/AbstractMapBasedMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Supplier.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/Constraint.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/MapDifference.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collection/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/InputSupplier.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Suppliers.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/LexicographicalOrdering.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/MapConstraints.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/SortedSetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/eventbus/Subscribe.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ImmutableList.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingQueue.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/RegularImmutableSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/AbstractMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/eventbus/SubscriberFindingStrategy.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/SetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/AbstractListMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ReverseNaturalOrdering.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/Resources.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/Flushables.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/EmptyImmutableSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingCollection.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/eventbus/SynchronizedEventSubscriber.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/Objects.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/EnumBiMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/Sets.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/EmptyContiguousSet.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/base/Preconditions.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/Interners.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/HashBiMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

Found in path(s):
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/SparseImmutableTable.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/MapMakerInternalMap.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/Cut.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/MapMaker.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/RegularImmutableTable.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/DenseImmutableTable.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Longs.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/CheckedFuture.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/ImmutableListMultimap.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/StandardTable.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/Serialization.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Chars.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/ImmutableEntry.java"
"/opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/net/InetAddresstes.java"
jar/jersey/repackaged/com/google/common/collection/StandardRowSortedTable.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Shorts.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ImmutableMapKeySet.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/UnmodifiableIterator.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Booleans.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableBiMap.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableListMultimap.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/PeeekingIterator.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/Table.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/CollectPreconditions.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/CollectPreconditions.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ImmutableMultiset.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableSortedSet.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ImmutableMapEntrySet.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/thirdparty/publicsuffix/TrieParser.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collect/HashBasedTable.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableSortedSet.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ImmutableMapKeySet.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/UnmodifiableIterator.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Shorts.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/Booleans.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableBiMap.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/EmptyImmutableListMultimap.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/PeeekingIterator.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/Table.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/CollectPreconditions.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/base/Stopwatch.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/escape/Escape.java
* /opt/cola/permits/113613812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/HashBasedTable.java

No license file was found, but licenses were detected in source scan.

/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at

Open Source Used In DNAC Platform Arica  11313
* http://creativecommons.org/publicdomain/zero/1.0/
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/cache/Striped64.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/util/concurrent/AtomicDoubleArray.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2005 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/reflect/Reflection.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
 * This method was written by Doug Lea with assistance from members of JCP
 * JSR-166 Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 * 
 * As of 2010/06/11, this method is identical to the (package private) hash
 * method in OpenJDK 7’s java.util.HashMap class.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/AbstractRangeSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/Count.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/GeneralRange.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/SortedIterables.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collect/ImmutableSortedMultisetFauxverideShim.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ForwardingSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/ImmutableSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/RangeSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/primitives/UnsignedLong.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/collection/SortedMultisets.java
No license file was found, but licenses were detected in source scan.
* the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/FunctionalEquivalence.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/*
* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
* following header:
* 
* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
* hereby disclaims copyright to this source code.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
*
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ImmutableSortedAsList.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/util/concurrent/ListenerCallQueue.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/MoreObjects.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS-IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/GenericMapMaker.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/Hashing.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * This following method is a modified version of one found in
 * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
 * which contained the following notice:
 * 
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 * Other contributors include Andrew Wright, Jeffrey Hayes,
 * Pat Fisher, Mike Judd.
 */

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/cache/Weigher.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/math/MathPreconditions.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collection/FilteredSetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collection/ImmutableEnumMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collection/DescendingImmutableSortedSet.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/collection/ForwardingNavigableMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/escape/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/html/package-info.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
  jar/jersey/repackaged/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-
jar/jersey/repackaged/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/RangeMap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/FilteredEntrySetMultimap.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/ForwardingBlockingDeque.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/AbstractInvitationHandler.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/TransformedListIterator.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/Closer.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1136113812_1613679412.19/0/jersey-guava-2-25-1-sources-4-jar/jersey/repackaged/com/google/common/collect/TreeRangeMap.java
1.2382 source-map-resolve 0.5.3 0.5.3

1.2382.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014, 2015, 2016, 2017, 2018, 2019 Simon Lydell
Copyright (c) 2019 ZHAO Jinxiang

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
turned to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2383 objenesis 2.6

1.2383.1 Available under license:

// ------------------------------------------------------------------
// NOTICE file corresponding to the section 4d of The Apache License,
// Version 2.0, in this case for Objenesis
// ------------------------------------------------------------------

Objenesis
Copyright 2006-2017 Joe Walnes, Henri Tremblay, Leonardo Mesquita

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2384 @babel/plugin-syntax-class-properties

7.12.13

1.2384.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2385 mkdirp 0.5.1 0.5.1

1.2385.1 Available under license :

Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2386 config 1.0.0

1.2386.1 Available under license:

The MIT License (MIT)
Copyright 2015 Hassan Khan <contact@hassankhan.me>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2387 hoek 2.16.3 2.16.3

1.2387.1 Available under license:

Copyright (c) 2011-2014, Walmart and other contributors.
Copyright (c) 2011, Yahoo Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The names of any contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/hapi/graphs/contributors

Portions of this project were initially based on the Yahoo! Inc. Postmile project, published at https://github.com/yahoo/postmile.

1.2388 resourcelocator 1.0.1

1.2388.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.
3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customary used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of
the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES
FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying
it may be distributed under the terms of this General Public License. The "Program", below, refers to any such
program or work, and a "work based on the Program" means either the Program or any derivative work under
copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications
and/or translated into another language. (Herein after, translation is included without limitation in the term
"modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its
scope. The act of running the Program is not restricted, and the output from the Program is covered only if its
contents constitute a work based on the Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium,
provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the
Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any
change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this
License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started
running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the
section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2
Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2389 hbase-client 2.4.2

1.2389.1 Available under license :

Apache HBase - Client
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2390 sway 2.0.6 2.0.6
1.2390.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Apigee Corporation
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2391 http-errors 1.7.2 1.7.2

1.2391.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong me@jongleberry.com
Copyright (c) 2016 Douglas Christopher Wilson doug@somethingdoug.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2392 dayjs 1.9.3 1.9.3
1.2392.1 Available under license:
MIT License

Copyright (c) 2018-present, iamkun

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2393 hbase-hadoop-compat 2.4.2
1.2393.1 Available under license:
Apache HBase - Hadoop Two Compatibility
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2394 @babel/plugin-transform-typeof-symbol 7.16.0

1.2394.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2395 has-bigints 1.0.1

1.2395.1 Available under license:

MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.2396 batik-gui-util 1.13

1.2396.1 Available under license:

Apache Batik
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software's
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2397 fragment-cache 0.2.1 0.2.1

1.2397.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2398 combined-stream 1.0.8 1.0.8

1.2398.1 Available under license :

Copyright (c) 2011 Debuggable Limited <felix@debuggable.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2399 inherits 2.0.4 2.0.4

1.2399.1 Available under license :

The ISC License

Copyright (c) Isaac Z. Schlueter
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

**1.2400 istanbul-lib-processinfo 2.0.2**

**1.2400.1 Available under license:**
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

**1.2401 babel-plugin-transform-member-expression-literals 7.16.0**

**1.2401.1 Available under license:**
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2402 @babel/plugin-syntax-private-
property-in-object 7.14.5
1.2402.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2403 commons-io 1.3.2
1.2403.1 Available under license:
Apache Jakarta Commons IO
Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2404 color-name 1.1.3 1.1.3
1.2404.1 Available under license :

The MIT License (MIT)
Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2405 file-entry-cache 2.0.0
1.2405.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Roy Riojas

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2406 source-map-support 0.4.18

1.2406.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Evan Wallace

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2407 jline-style 3.12.1
1.2407.1 Available under license:

No license file was found, but licenses were detected in source scan.

2016, the original author or authors.

This software is distributable under the BSD license. See the terms of the BSD license in the documentation provided with this software.

https://opensource.org/licenses/BSD

Found in path(s):
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/META-INF/maven/org.jline/jline-style/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2018, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the BSD license in the documentation provided with this software.
 *
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/Styler.java
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/StyleSource.java
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/StyleExpression.java
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/StyleFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2017, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the BSD license in the documentation provided with this software.
 *
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/InterpolationHelper.java
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/MemoryStyleSource.java
*/ /opt/cola/permits/1114600388_1607534029.29/0/jline-style-3-12-1-sources-jar/org/jline/style/StyleBundleInvocationHandler.java
1.2408 xtend 2.1.2 2.1.2

1.2408.1 Available under license:

Copyright (c) 2012 Raynos.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2409 babel-plugin-transform-es2015-computed-properties 6.24.1

1.2409.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2410 joda-time 2.10.5

1.2410.1 Available under license:

==============================================================================
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
==============================================================================

This product includes software developed by Joda.org (https://www.joda.org/).
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2411.hbase-zookeeper 2.4.2
1.2411.1 Available under license:

Apache HBase - Zookeeper
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets ""[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2412 asm-commons 7.3.1
1.2412.1 Available under license :
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new {@link LocalVariablesSorter}. <i>Subclasses must not use this constructor</i>.
 * Instead, they must use the {@link #LocalVariablesSorter(int, int, String, MethodVisitor)}
 * version.
 *
 * @param access access flags of the adapted method.
 * @param descriptor the method's descriptor (see {@link Type}).
 * @param methodVisitor the method visitor to which this adapter delegates calls.
 * @throws IllegalStateException if a subclass calls this constructor.
 */

Found in path(s):
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/LocalVariablesSorter.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom
* All rights reserved.
 *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* *
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/package.html
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new SerialVersionUIDAdder. <i>Subclasses must not use this
 * constructor</i>. Instead, they must use the SerialVersionUIDAdder(int, ClassVisitor)
 * version.
 *
 * @param classVisitor a ClassVisitor to which this visitor will delegate calls.
 * @throws IllegalStateException If a subclass calls this constructor.
 */

Found in path(s):
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/SerialVersionUIDAdder.java

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/StaticInitMerger.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/ModuleResolutionAttribute.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/AnalyzerAdapter.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
jar/org/objectweb/asm/commons/ClassRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/SignatureRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/Method.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/AdviceAdapter.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/Remapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/MethodRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/SimpleRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/AnnotationRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/ModuleTargetAttribute.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/FieldRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/CodeSizeEvaluator.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/ModuleHashesAttribute.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/InstructionAdapter.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/TryCatchBlockSorter.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/ModuleRemapper.java
* /opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-jar/org/objectweb/asm/commons/RecordComponentRemapper.java
No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
* Constructs a new `@link GeneratorAdapter`. *Subclasses must not use this constructor*.
* Instead, they must use the `@link #GeneratorAdapter(int, MethodVisitor, int, String, String)`
* version.
*
* `@param methodVisitor` the method visitor to which this adapter delegates calls.
* `@param access` the method's access flags (see `@link Opcodes`).
* `@param name` the method's name.
* `@param descriptor` the method's descriptor (see `@link Type`).
* `@throws IllegalStateException` if a subclass calls this constructor.
*
Found in path(s):
* `/opt/cola/permits/1135907511_1613628615.48/0/asm-commons-7-3-1-sources-1-
  jar/org/objectweb/asm/commons/GeneratorAdapter.java`

---

1.2413 libcroco 0.6.11 1

1.2413.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

---

Open Source Used In DNAC Platform Arica 11384
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.
Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must
be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.2414 ejs 2.6.2 2.6.2
1.2414.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2415 yargs 13.3.2

1.2415.1 Available under license :

Copyright 2010 James Halliday (mail@substack.net)
Modified work Copyright 2014 Contributors (ben@npmjs.com)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2416 mockito 2.23.4
1.2416.1 Available under license :
   The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2417 proxy-addr 2.0.4 2.0.4
1.2417.1 Available under license :
   (The MIT License)

Copyright (c) 2014-2016 Douglas Christopher Wilson
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2418 is-accessor-descriptor 1.0.0

1.2418.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2419 is-installed-globally 0.4.0
1.2419.1 Available under license:
   MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2420 jsreport-debug 2.1.3 2.1.3
1.2420.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2421 caching-transform 4.0.0

1.2421.1 Available under license:

The MIT License (MIT)

Copyright (c) James Talmage <james@talmage.io> (github.com/jamestalmage)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2422 responselike 1.0.2

1.2422.1 Available under license:

Copyright (c) 2017 Luke Childs

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1.2423 har-schema 2.0.0 2.0.0

1.2423.1 Available under license:

Copyright (c) 2015, Ahmad Nassri <ahmad@ahmadnassri.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2424 jsesc 1.3.0 1.3.0

1.2424.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2425 define-properties 1.1.3 1.1.3

1.2425.1 Available under license :
The MIT License (MIT)

Copyright (C) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2426 optional-require 1.0.3 1.0.3

1.2426.1 Available under license :
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with
that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   a. You must give any other recipients of the Work or Derivative Works a copy of this License; and

   b. You must cause any modified files to carry prominent notices stating that You changed the files; and

   c. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   d. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for
informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2427 babel-traverse 6.26.0 6.26.0

1.2427.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2428 ant 1.10.9
1.2428.1 Available under license:

/*
 * Apache License
 * Version 2.0, January 2004
 * http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction,
 * and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by
 * the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all
 * other entities that control, are controlled by, or are under common
 * control with that entity. For the purposes of this definition,
 * "control" means (i) the power, direct or indirect, to cause the
 * direction or management of such entity, whether by contract or
 * otherwise, or (ii) ownership of fifty percent (50%) or more of the
 * outstanding shares, or (iii) beneficial ownership of such entity.
 *
 * "You" (or "Your") shall mean an individual or Legal Entity
 * exercising permissions granted by this License.
 *
 * "Source" form shall mean the preferred form for making modifications,
 * including but not limited to software source code, documentation
 * source, and configuration files.
 *
 * "Object" form shall mean any form resulting from mechanical
 * transformation or translation of a Source form, including but
 * not limited to compiled object code, generated documentation,
 * and conversions to other media types.
 *
 * "Work" shall mean the work of authorship, whether in Source or
 * Object form, made available under the License, as indicated by a
 * copyright notice that is included in or attached to the work
 * (an example is provided in the Appendix below).
 *
 * "Derivative Works" shall mean any work, whether in Source or Object
 * form, that is based on (or derived from) the Work and for which the
 * editorial revisions, annotations, elaborations, or other modifications
 * represent, as a whole, an original work of authorship. For the purposes
 * of this License, Derivative Works shall not include works that remain
 * separable from, or merely link (or bind by name) to the interfaces of,
 * the Work and Derivative Works thereof.
 */
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

W3C SOFTWARE NOTICE AND LICENSE
https://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission.
Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pd

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05
Apache Ant
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

The <sync> task is based on code Copyright (c) 2002, Landmark Graphics Corp that has been kindly donated to the Apache Software Foundation.

1.2429 dom-serializer 0.1.0 0.1.0
1.2429.1 Available under license:

License

(The MIT License)

Copyright (c) 2014 The cheeriojs contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2430 stax-ex 1.8

1.2431 jasper-runtime 5.5.12

1.2431.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are
under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and You must cause any modified files to carry prominent notices stating that You changed the files; and You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2432 v8flags 2.1.1
1.2432.1 Available under license :
Copyright (c) 2014 Tyler Kellen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2433.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2434 blob 0.0.4 0.0.4
1.2434.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2435 extend-shallow 2.0.1 2.0.1
1.2435.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2436 javax-jdo 3.2.0-m3

1.2436.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
END OF TERMS AND CONDITIONS

Copyright 2015-2015 DataNucleus

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2437 chalk 1.1.3
1.2437.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2438 react-tagsinput 3.19.0 3.19.0
1.2438.1 Available under license:

The MIT License

Copyright (c) 2015 Ola Holmström <olaholmstrom+github@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2439 is-regex 1.1.2

1.2439.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2440 is-descriptor 0.1.6
1.2440.1 Available under license: MIT

1.2441 is-symbol 1.0.3
1.2441.1 Available under license: The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2442 async 0.9.2 0.9.2
1.2442.1 Available under license: Copyright (c) 2010-2014 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2443 reftools 1.1.8 1.1.8

1.2443.1 Available under license:

BSD 3-Clause License

Copyright (c) 2017, Mike Ralphson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2444 jmockit 1.47
1.2444.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006 JMockit developers
 * This file is subject to the terms of the MIT license (see LICENSE.txt).
 */

Found in path(s):
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/util/Utilities.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/ParameterTypeRedefinitions.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/MockedType.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/TestedParameters.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/integration/full/TestConversation.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/integration/testng/package-info.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/full/FullInjection.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/invocation/ArgumentValuesAndMatchersWithVarargs.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/modification/ClassSelection.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/invocation/MultiValuedConversion.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/MockedBridge.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/reporting/sourceFiles/FileCoverageReport.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/argumentMatching/EqualityMatcher.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/ArgumentMatcher.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/TypeRedefinitions.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/argumentMatching/LenientEqualityMatcher.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/modification/VisitInterruptedException.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/TypeRedefinitions.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/modification/VisitInterruptedException.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/reporting/SourceFiles.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/Tested.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/invocation/InvocationResults.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/InstanceFactory.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/CoverageCheck.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/ClassFile.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/full/ServletDependencies.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/MultiValuedProvider.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/CoverageCheck.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/CoverageCheck.java
Open Source Used In DNAC Platform Arica 11432

* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/integration/springframework/BeanLookup.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/lines/BranchCoverageData.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/InjectionProvider.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/testRedundancy/JUnitListener.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/faking/FakeClasses.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/reflection/GenericTypeReflection.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/RecordPhase.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/state/MockedTypeCascade.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/mocking/FieldTypeRedefinitions.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/argumentMatching/StringContainmentMatcher.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/injection/TestedObject.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/reporting/lineCoverage/LineCoverageFormatter.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/util/DefaultValues.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/reporting/DelegatedResult.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/coverage/reporting/lineCoverage/LineCoverageFormatter.java
* /opt/cola/permits/1202446415_1631568333.62/0/jmockit-1-47-sources-jar/mockit/internal/expectations/invo
1.2445 snapdragon 0.8.2 0.8.2
1.2445.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2446 is-absolute 1.0.0 1.0.0
1.2446.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.
Copyright (c) 2009-2014, TJ Holowaychuk

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2447 object-keys 0.4.0 0.4.0

1.2447.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2448 jackson-datatype-joda 2.12.1

1.2448.1 Available under license:

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
1.2449 netty 4.1.56.Final
1.2449.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Copyright 2014 Twitter, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/HpackDynamicTable.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/HpackHuffmanDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/HpackHeaderField.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/HpackEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/HpackUtil.java
*/
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2012 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

//The MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software,
//furnished to do so, subject to the following conditions:
//The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

No license file was found, but licenses were detected in source scan.

* Copyright 2013 The Netty Project

* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/locale/http/cors/CorsHandler.java

No license file was found, but licenses were detected in source scan.

* Copyright 2013 The Netty Project

* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/locale/http/cors/CorsHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/locale/http/cors/CorsConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http/cors/OkResponseHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketClientProtocolHandshakeHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketFrameDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/ChannelId.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/internal/EmptyArrays.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketClientProtocolHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/memcache/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/concurrent/AbstractEventExecutorGroup.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/xml/XmlFrameDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/concurrent/ImmediateExecutor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/memcache/binary/DefaultFullBinaryMemcacheResponse.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/concurrent/DefaultPromise.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/udt/UdtChannelOption.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/socks/SocksSubnegotiationVersion.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/memcache/binary/DefaultBinaryMemcacheResponse.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
Open Source Used in DNAC Platform Arica 11443

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/ConcurrentSet.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdySessionStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdySynReplyFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyDataFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/DefaultSpdyHeaders.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/DefaultSpdyDataFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/DefaultSpdyWindowUpdateFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/NoOpTypeParameterMatcher.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/NoOpTypeParameterMatcher.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/AbstractFuture.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/ChannelHandlerAdapter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/socket/oio/DefaultOioServerSocketChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/socket/oio/DefaultOioServerSocketChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/FullBinaryMemcacheRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheContent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/BinaryMemcacheMessage.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/FullHttpResponse.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/WebSocketFrameEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socks/SocksAddressType.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socks/DefaultSocksAddressType.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socks/DefaultSocksServerAddressType.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheCommand.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/MessageSizeEstimator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyWindowUpdateFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/NoOpTypeParameterMatcher.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheCommand.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/memcache/binary/DefaultMemcacheRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyDataFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyHeaderBlockZlibEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/concurrent/Future.java
Open Source Used In DNAC Platform Arica

11447
Open Source Used In DNAC Platform Arica 11448

jar/io/netty/handler/codec/http/FullHttpRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksCmdType.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/spdy/SpdyPingFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/spdy/SpdyStreamFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/memcache/binary/DefaultFullBinaryMemcacheRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/memcache/binary/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/memcache/binary/AbstractBinaryMemcacheEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/memcache/binary/AbstractBinaryMemcacheObject.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/spdy/DefaultSpdyDataFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/spdy/DefaultSpdyHeadersFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/spdy/SpdyStreamFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksCmdStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksAuthStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/socket/oio/OioSocketChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/sv5/Socks5PasswordAuthStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksCmdStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksAuthStatus.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/worldclock/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/socket/SocketChannelFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/socket/OioSocketChannelConfig.java
jar/io/netty/handler/codec/memcache/binary/BinaryMemcacheOpCodes.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/nio/SelectedSelectionKeySet.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/ComposedLastHttpContent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketProtocolHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/memcache/binary/BinaryMemcacheResponse.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/concurrent/GenericProgressiveFutureListener.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// (BSD License: https://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/websocketx/WebSocket07FrameDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/websocketx/WebSocket13FrameDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/websocketx/WebSocket08FrameDecoder.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/KmpSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/http2/helloworld/client/Http2ClientFrameInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/http2/helloworld/client/Http2Client.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/AbstractMultiSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/MultiSearchProcessor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/SearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/AbstractMultiSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/AbstractMultiSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/BitapSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/AhoCorasicSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/SearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/MultiSearchProcessor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/SearchProcessor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/MultiSearchProcessorFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/SearchProcessor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/search/MultiSearchProcessorFactory.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2018 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/internal/tnative/CertificateCallback.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2020 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/stomp/websocket/StompWebSocketFrameEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/stomp/websocket/StompVersion.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/SslHandshakeTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/address/ResolveAddressHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/socket/DuplexChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttUnsubAckPayload.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/NetUtilInitializations.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/address/ResolveAddressHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/stomp/websocket/StompSubscription.java
Open Source Used In DNAC Platform Arica 11453

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/logging/ByteBufFormat.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/dns/dot/DoTClient.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/util/NetUtilSubstitutions.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/stomp/websocket/StompWebSocketProtocolCodec.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/mqtt/MqttReasonCodeAndPropertiesVariableHeader.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/haproxy/HAProxyServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/StacklessClosedChannelException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http/multipart/DeleteFileOnExitHook.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/mqtt/MqttPubReplyMessageVariableHeader.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/search/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/LongPriorityQueue.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/LongLongHashMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/stomp/websocket/StompWebSocketChatServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/ssl/SslClosedEngineException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/haproxy/HAProxyMessageEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/ipfilter/IpSubnetFilterRuleComparator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/dns/udp/DnsClient.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/haproxy/HAProxyHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/ssl/SslClosedEngineException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/stomp/websocket/StompWebSocketChatServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http/websocketx/WebSocketClientHandshakeException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/stomp/websocket/StompWebSocketClientPageHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

// https://tools.ietf.org/html/rfc7540#section-8.1.2.2 makes a special exception for TE

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 */
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
  * "License"); you may not use this file except in compliance with the License. You may obtain a
  * copy of the License at:
  * 
  * https://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software distributed under the License
  * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
  * express
  * or implied. See the License for the specific language governing permissions and limitations under
  * the License.
  */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/Http2ConnectionPrefaceAndSettingsFrameWrittenEvent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/AbstractCoalescingBufferQueue.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/PseudoRandomFunction.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/SslMasterKeyHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionFilter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/Http2MultiplexHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/Buffer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/OpenSslTrustManager.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/OpenSslTlsv13X509ExtendedTrustManager.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/AuthoritativeDnsServerCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/NoopAuthoritativeDnsServerCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/DirContextUtils.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/DnsAddressResolveContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/DnsAddressResolveContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/DnsAddressResolveContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/AuthoritativeDnsServerCacheAdapter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/resolver/dns/DnsCnameCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslSession.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/resolver/dns/DnsRecordResolveContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/resolver/dns/DefaultAuthoritativeDnsServerCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/SimpleUserEventChannelHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/socket/nio/NioChannelOption.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslX509TrustManagerWrapper.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslKeyMaterialProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslX509KeyManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslCachingX509KeyManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslX509KeyManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslX509KeyManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/SuppressJava6Requirement.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/resolver/dns/DnsAddressDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/OpenSslX509TrustManagerWrapper.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/haproxy/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websockets/extensions/compression/WebSocketServerCompressionHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/Epoll.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/haproxy/package-info.java

Open Source Used In DNAC Platform Arica 11460
Open Source Used In DNAC Platform Arica 11464

jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketServerExtensionHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/JdkApplicationProtocolNegotiator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/mqtt/MessageIdVariableHeader.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/compression/FastLzFrameEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/util/ThreadLocalInsecureRandom.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/mqtt/UnsubAckMessage.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/compression/Bzip2Constants.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2OutboundFrameLogger.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/socksx/v5/Socks5InitialRequestDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/concurrent/PromiseAggregator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/rtp/RtpHeaderValues.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2FrameLogger.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/xml/XmlProcessingInstruction.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/util/FingerprintTrustManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/example/spdy/client/SpdyClientInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/compression/FastLz.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketClientExtensionHandshaker.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/IntegerHolder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/compression/LzfDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/ssl/JdkSslServerContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
Open Source Used In DNAC Platform Arica 11467

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/Mapping.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/compression/FastLzFrameDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/ResourceLeakHint.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/proxy/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http2/helloworld/server/HelloWorldHttp1Handler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/stomp/StompSubframeDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/xml/XmlElementEnd.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/stomp/StompSubframe.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/EpollSocketChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/HttpChunkedInput.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/proxy/ProxyHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/haproxy/HAProxyMessageDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/compression/Bzip2Decoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/compression/Lz4Constants.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/AbstractEpollChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/haproxy/HAProxyProtocolException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/EpollChannelOption.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websockets/extensions/compression/PerMessageDeflateEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/spdy/SpdyFrameEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttMessageFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/compression/SnappyFramedDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/OpenSslServerContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ApplicationProtocolUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http/websocketx/client/WebSocketClient.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/compression/Bzip2DivSufSort.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socksx/v5/Socks5PasswordAuthResponseDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttConnectMessage.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/socksx/v5/Socks5ServerEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/socksx/v5/Socks5PasswordAuthRequestEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/extensions/compression/PerMessageDeflateClientExtensionHandshaker.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/proxy/HttpProxyHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/http/websocketx/benchmarkserver/WebSocketServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/example/memcache/binary/MemcacheClientHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/ipfilter/IpFilterRule.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/compression/Bzip2MoveToFrontTable.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/compression/Bzip2HuffmanAllocator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/compression/Bzip2Rand.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/http2/helloworld/server/Http2ServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/http/HttpHeadersEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/mqtt/MqttSubscribeMessage.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/mqtt/MqttTopicSubscription.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/mqtt/SniHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/mqtt/NoopAddressResolver.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codecs/mqtt/NoopAddressResolver.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/EmbeddedChannelId.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/EmbeddedChannelId.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/embedded/OpenSsContext.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/PooledHeapByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// (BSD License: https://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocket13FrameEncoder.java
*/
jar/io/netty/handler/codec/http/websocketx/WebSocket07FrameEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocket08FrameEncoder.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/HttpServerUpgradeHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2ConnectionHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2DataWriter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2ConnectionEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/collection/ShortObjectMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2Connection.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/EmptyHttp2Headers.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/DefaultHttp2ConnectionEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/DefaultHttp2HeadersDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2DataWriter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/collection/CharObjectMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2ConnectionEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/DefaultHttp2HeadersDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http2/Http2ConnectionEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/collection/LongObjectHashMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/DefaultHeaders.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/EmptyHeaders.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2ConnectionAdapter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2EventAdapter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/collection/LongObjectMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FlowController.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameWriter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameListener.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameTypes.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2LifecycleManager.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2OrHttpHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2SettingsHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2ClientInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/collection/ByteCollections.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2 coerce/Http2Connection.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameReader.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FlowController.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2HeaderEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2SettingsEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/ObjectUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2Settings.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 */
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4.1.56-final-sources-
  jar/io/netty/handler/traffic/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4.1.56-final-sources-
  jar/io/netty/handler/traffic/GlobalTrafficShapingHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4.1.56-final-sources-
  jar/io/netty/handler/traffic/TrafficCounter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4.1.56-final-sources-
  jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4.1.56-final-sources-
  jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.
/*
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/traffic/AbstractTrafficShapingHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpServerChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/nio/NioSctpChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/nio/NioSctpServerChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/DefaultSctpChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpServerChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpNotificationHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpMessage.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/sctp/SctpChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/worldclock/WorldClockProtocol.proto

No license file was found, but licenses were detected in source scan.

/
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/redis/FixedRedisMessagePool.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/redis/RedisCodecUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/redis/RedisMessageType.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/redis/RedisMessagePool.java
/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 * *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging/FormattingTuple.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/logging/MessageFormatter.java
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/DefaultChannelHandlerContext.java
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/CleanerJava6.java
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/buffer/UnsafeHeapSwappedByteBuf.java
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/concurrent/FastThreadLocalThread.java
 * /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/buffer/UnsafeDirectSwappedByteBuf.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/* Adaptation of https://bjoern.hoehrmann.de/utf-8/decoder/dfa/
 *
 * Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 * and associated documentation files (the "Software"), to deal in the Software without restriction,
 * including without limitation the rights to use, copy, modify, merge, publish, distribute,
 * sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or
 * substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING, BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
 * PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
 * FOR ANY CLAIM,
 * DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
 * ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
 * SOFTWARE.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/websocketx/Utf8Validator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/
/*
 * Licensed to the Apache Software Foundation (ASF) under one or more

* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/internal/tcnative/SSL.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/internal/tcnative/SSLContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/internal/tcnative/Buffer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/internal/tcnative/Library.java

No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/shaded/org/jctools/maps/NonBlockingSetInt.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/shaded/org/jctools/util/UnsafeLongArrayAccess.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/shaded/org/jctools/queues/IndexedQueueSizeUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
Open Source Used In DNAC Platform Arts 11487
/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/internal/tnnative/CertificateVerifier.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 *
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging/JdkLogger.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging/Log4JLogger.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging/CommonsLogger.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging/InternalLogger.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~  https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/META-
INF/maven/io.netty/netty-all/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *  https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/sctp/DefaultSctpServerChannelConfig.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/resolver/dns/DnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/SniCompletionEvent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/PooledByteBufAllocatorMetric.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/resolver/dns/NoopDnsQueryLifecycleObserverFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/OptionalSslHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/OcspClientHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/uptime/UptimeServerHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/SingletonDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/DelegatingChannelPromiseNotifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/hand...
Open Source Used In DNAC Platform Arica 11494

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2MultiplexCodecBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/ReadOnlyHttpHeaders.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/mqtt/MqttMessageBuilders.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameCodecBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2UnknownFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/LoggingDnsQueryLifecycleObserver.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/socket/ChannelInputShutdownReadComplete.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/Conscrypt.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ResolvedAddressTypes.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/WebSocketScheme.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/AbstractSniHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/JdkAlpnSslUtils.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameCodecBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/uptime/UptimeServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/SuppressForbidden.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/JdkAlpnSslEngine.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2UnknownFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2FrameCodecBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DnsNameResolverTimeoutException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/SequentialDnsServerAddressStreamProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/ReflectionUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/BiDnsQueryLifecycleObserverFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/UnixResolverDnsServerAddressStreamProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/socket/oio/DefaultOioDatagramChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/UncheckedBooleanSupplier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/PriorityQueue.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/bootstrap/FailedChannel.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/proxy/HexDumpProxyBackendHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/qotm/QuoteOfTheMomentServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/CorruptedFrameException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketHandshakeException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/buffer/ReadOnlyByteBuf.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspRequestDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/serialization/WeakReferenceMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/socket/DefaultDatagramChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/timeout/ReadTimeoutHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/CorruptedFrameException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/socket/ChannelInputShutdownEvent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/http/file/HttpStaticFileServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketHandshakeException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/socket/DefaultDatagramChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/udt/echo/rendezvousBytes/ByteEchoPeerTwo.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/rtsp/RtspRequestDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/serialization/WeakReferenceMap.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/ssl/package-info.java

Open Source Used In DNAC Platform Arica  11496
jar/io/netty/handler/ss/util/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/multipart/FileUpload.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/example/http/snoop/HttpSnoopClientInitializer.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/logging.Slf4jLoggerFactory.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/compression/DecompressionException.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/multipart/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/protobuf/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util_INTERNAL/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/socksx/v5/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/bytes/ByteArrayDecoder.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/multipart/package-info.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/serialization/CompactObjectOutputStream.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/serialization/CompatibleObjectEncoder.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/socksx/v5/DefaultSocks5InitialResponse.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/channel/sctp/oio/OioSctpChannel.java
* /opt/cola/permits/111902601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/logging/LoggingHandler.java
jar/io/netty/handler/codec/bytes/ByteArrayEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/marshalling/DefaultMarshallerProvider.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/socksx/v5/DefaultSocks5InitialRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/echo/EchoClientHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/concurrent/EventExecutor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/socksx/v5/DefaultSocks5PasswordAuthResponse.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/serialization/ObjectEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/factorial/BigIntegerDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/echo/EchoClient.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/base64/Base64Encoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/udt/echo/rendezvous/MsgEchoPeerHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/channel/CompleteChannelFuture.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/http/upload/HttpUploadServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/multipart/MemoryAttribute.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/http/file/HttpStaticFileServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/spdy/SpdySession.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/HttpContent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/spdy/SpdySession.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/HttpContent.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http/HttpStaticFileServerInitializer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/example/portunification/PortUnificationServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/multipart/HttpPostBodyUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
Open Source Used In DNAC Platform Arica 11503

info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/securechat/SecureChatClient.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/ChannelException.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/group/DefaultChannelGroupFuture.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/RecvByteBufAllocator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/ContinuationWebSocketFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/codec/serialization/ObjectEncoderOutputStream.java
Open Source Used In DNAC Platform Arica 11504
Open Source Used In DNAC Platform Arica 11509
Open Source Used In DNAC Platform Arica 11515
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// Try the OpenJDK's proprietary implementation.

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io.netty.handler/ssl/util/SelfSignedCertificate.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * https://creativecommons.org/publicdomain/zero/1.0/
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
Copyright 2012 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

A decoder that splits the received `ByteBuf`s dynamically by the value of the length field in the message. It is particularly useful when you decode a binary message which has an integer header field that represents the length of the message body or the whole message.

* [2 bytes length field at offset 0, do not strip header](h3>

The value of the length field in this example is `12 (0x0C)` which represents the length of "HELLO, WORLD". By default, the decoder assumes that the length field represents the number of the bytes that follows the length field. Therefore, it can be decoded with the simplistic parameter combination.

```
| lengthFieldOffset = <b>0</b> |
| lengthFieldLength = <b>2</b> |
```

**BEFORE DECODE (14 bytes) | AFTER DECODE (14 bytes)**

```
<table>
<thead>
<tr>
<th>Length</th>
<th>Actual Content</th>
<th>Length</th>
<th>Actual Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000C</td>
<td>&quot;HELLO, WORLD&quot;</td>
<td>0x000C</td>
<td>&quot;HELLO, WORLD&quot;</td>
</tr>
</tbody>
</table>
```

</pre>
Because we can get the length of the content by calling
[@link ByteBuf#readableBytes()], you might want to strip the length
field by specifying <tt>initialBytesToStrip</tt>. In this example, we
specified <tt>2</tt>, that is same with the length of the length field, to
strip the first two bytes.

```java
lengthFieldOffset   = 0
lengthFieldLength   = 2
lengthAdjustment    = 0
initialBytesToStrip = 2 (= the length of the Length field)
```

BEFORE DECODE (14 bytes)          AFTER DECODE (12 bytes)
+--------+----------------+      +----------------+
| Length | Actual Content |----->| Actual Content |
| 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
+--------+----------------+      +----------------+

In most cases, the length field represents the length of the message body
only, as shown in the previous examples. However, in some protocols, the
length field represents the length of the whole message, including the
message header. In such a case, we specify a non-zero
<tt>lengthAdjustment</tt>. Because the length value in this example message
is always greater than the body length by <tt>2</tt>, we specify <tt>-2</tt> as <tt>lengthAdjustment</tt> for compensation.

```java
lengthFieldOffset   = 0
lengthFieldLength   = 2
lengthAdjustment    = -2 (= the length of the Length field)
initialBytesToStrip = 0
```

BEFORE DECODE (14 bytes)          AFTER DECODE (14 bytes)
+--------+----------------+      +--------+----------------+
| Length | Actual Content |----->| Length | Actual Content |
| 0x000E | "HELLO, WORLD" |      | 0x000E | "HELLO, WORLD" |
+--------+----------------+      +--------+----------------+

The following message is a simple variation of the first example. An extra
header value is prepended to the message. <tt>lengthAdjustment</tt> is zero
again because the decoder always takes the length of the prepended data into
account during frame length calculation.

* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>

* This is an advanced example that shows the case where there is an extra
header between the length field and the message body. You have to specify a
positive <tt>lengthAdjustment</tt> so that the decoder counts the extra
header into the frame length calculation.

* <pre>
  * lengthFieldOffset   = 0
  * lengthFieldLength   = 3
  * lengthAdjustment    = 2 (= the length of Header 1)
  * initialBytesToStrip = 0
  *
  * BEFORE DECODE (17 bytes)                  AFTER DECODE (17 bytes)
  +-------------------------------------------+-------------------------------------------+
  | Header 1 | Length | Actual Content |----->| Header 1 | Length | Actual Content |
  | 0xCAFE | 0x00000C | "HELLO, WORLD" |      | 0xCAFE | 0x00000C | "HELLO, WORLD" |
  +-------------------------------------------+-------------------------------------------+
  *
  </pre>

* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header,
strip the first header field and the length field</h3>

* This is a combination of all the examples above. There are the prepended
header before the length field and the extra header after the length field.
The prepended header affects the <tt>lengthFieldOffset</tt> and the extra
header affects the <tt>lengthAdjustment</tt>. We also specified a non-zero
<tt>initialBytesToStrip</tt> to strip the length field and the prepended
header from the frame. If you don't want to strip the prepended header, you
could specify <tt>0</tt> for <tt>initialBytesToSkip</tt>.

* <pre>
  * lengthFieldOffset   = 1 (= the length of HDR1)
  * lengthFieldLength   = 2
  * lengthAdjustment    = 1 (= the length of HDR2)
  *
  * BEFORE DECODE (17 bytes)                  AFTER DECODE (17 bytes)
  +-------------------------------------------+-------------------------------------------+
  | Length | Header 1 | Actual Content |----->| Length | Header 1 | Actual Content |
  | 0x00000C | 0xCAFE | "HELLO, WORLD" |      | 0x00000C | 0xCAFE | "HELLO, WORLD" |
  +-------------------------------------------+-------------------------------------------+
  *
  </pre>

*
* <b>initialBytesToStrip</b> = <b>3</b> (= the length of HDR1 + LEN)
*
* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +----------------------------------+                  +----------------------------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +----------------------------------+                  +----------------------------------+
* </pre>
*
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header,
* strip the first header field and the length field, the length field
* represents the length of the whole message</h3>
*
* Let's give another twist to the previous example. The only difference from
* the previous example is that the length field represents the length of the
* whole message instead of the message body, just like the third example.
* We have to count the length of HDR1 and Length into <tt>lengthAdjustment</tt>.
* Please note that we don't need to take the length of HDR2 into account
* because the length field already includes the whole header length.
* <pre>
* lengthFieldOffset   =  1
* lengthFieldLength   =  2
* <b>lengthAdjustment</b>    = <b>-3</b> (= the length of HDR1 + LEN, negative)
* <b>initialBytesToStrip</b> = <b> 3</b>
* 
* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +----------------------------------+                  +----------------------------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x0010 | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +----------------------------------+                  +----------------------------------+
* </pre>
* @see LengthFieldPrepender
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/CompositeNameResolver.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http2/tiles/Http1RequestHandler.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/pool/package-info.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/protobuf/ProtobufVarint32LengthFieldPrepender.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DatagramDnsResponseEncoder.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socksx/SocksPortUnificationServerHandler.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/EmptyHttpHeaders.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/DefaultCookie.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/DomainSocketReadMode.java
  */opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/FileDescriptor.java
Open Source Used In DNAC Platform Arica 11525

* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/RotationalDnsServerAddresses.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/PriorityQueueNode.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/CookieUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/OpenSslSessionTicketKey.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/HostsFileEntriesResolver.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DefaultDnsRawRecord.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/DefaultPriorityQueue.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/DomainSocketChannelConfig.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/SslContextBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http2/tiles/HttpServer.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DatagramDnsResponseDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/CookieEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/CookieUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/ClientCookieEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/UnsupportedValueConverter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/EpollRecvByteAllocatorHandle.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/dns/DnsQueryContextManager.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http2/tiles/Http2Server.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/UnixChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/CookieEncoder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/example/http2/helloworld/server/HelloWorldHttp2HandlerBuilder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/HttpExpectationFailedEvent.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/MaxMessagesRecvByteBufAllocator.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/AbstractHttp2ConnectionHandlerBuilder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/protobuf/ProtobufDecoder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/decoder/protobuf/ProtobufEncoderNano.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/cookie/Cookie.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DatagramDnsQueryDecoder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/util/AsyncMapping.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/resolver/DefaultHostsFileEntriesResolver.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/socksx/v5/Socks5AddressEncoder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/NativeInetAddress.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/dns/DefaultDnsResponse.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/epoll/EpollEventArray.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/epoll/EpollChannelConfig.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/decoder/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/decoder/ProtocolDetectionResult.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/sockets/v5/Socks5AddressEncoder.java
* /opt/cola/permits/1119092601_16144555572/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ApplicationProtocolNegotiationHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.
*/
/**
 * Set the \{@code TCP_MD5SIG\} option on the socket. See \{@code linux/tcp.h\} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
/**
 * Set the \{@code TCP_QUICKACK\} option on the socket.
 * See <a href="https://linux.die.net//man/7/tcp">TCP_QUICKACK</a> for more details.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/epoll/EpollSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2EmptyDataFrameListener.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2ControlFrameLimitEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2EmptyDataFrameConnectionDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2SettingsReceivedConsumer.java
No license file was found, but licenses were detected in source scan.

* Copyright 2012 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Handles an I/O event or intercepts an I/O operation, and forwards it to its next handler in its [link ChannelPipeline].

Sub-types

[@link ChannelHandler] itself does not provide many methods, but you usually have to implement one of its subtypes:

- [@link ChannelInboundHandler] to handle inbound I/O events, and
- [@link ChannelOutboundHandler] to handle outbound I/O operations.

Alternatively, the following adapter classes are provided for your convenience:

- [@link ChannelInboundHandlerAdapter] to handle inbound I/O events,
- [@link ChannelOutboundHandlerAdapter] to handle outbound I/O operations,
- [@link ChannelDuplexHandler] to handle both inbound and outbound events

For more information, please refer to the documentation of each subtype.

The context object

A [@link ChannelHandler] is provided with a [@link ChannelHandlerContext] object. A [@link ChannelHandler] is supposed to interact with the [@link ChannelPipeline] it belongs to via a context object. Using the context object, the [@link ChannelHandler] can pass events upstream or downstream, modify the pipeline dynamically, or store the information (using [@link AttributeKey])s) which is specific to the handler.

State management

Open Source Used In DNAC Platform Arica 11531
* A `@link ChannelHandler` often needs to store some stateful information.
* The simplest and recommended approach is to use member variables:
  * `<pre>`
  * public interface Message {
  *     // your methods here
  * }
  *
  * public class DataServerHandler extends `@link SimpleChannelInboundHandler<Message>` {  
  *     <b>private boolean loggedIn;</b>  
  * }
  *
  * `<h4>Using `@link AttributeKey`s</h4>`
  *
  * Although it's recommended to use member variables to store the state of a
  * handler, for some reason you might not want to create many handler instances.
In such a case, you can use [@link AttributeKey]s which is provided by
[@link ChannelHandlerContext]:
<pre>
public interface Message {
    // your methods here
}
</pre>

[@code @Sharable]
public class DataServerHandler extends [@link SimpleChannelInboundHandler]&lt;Message&gt; {
    private final [@link AttributeKey]&lt;[@link Boolean]&gt; auth =
    [@link AttributeKey#valueOf(String) AttributeKey.valueOf("auth")];

    @Override
    public void channelRead([@link ChannelHandlerContext] ctx, Message message) {
        Attribute&lt;Boolean&gt; attr = ctx.attr(auth);
        if (message instanceof LoginMessage) {
            authenticate((LoginMessage) o);
            attr.set(true);
        } else (message instanceof GetDataMessage) {
            if (Boolean.TRUE.equals(attr.get())) {
                ctx.writeAndFlush(fetchSecret((GetDataMessage) o));
            } else {
                fail();
            }
        }
    }

    ... ...
}</pre>

Now that the state of the handler is attached to the [@link ChannelHandlerContext], you can add the
same handler instance to different pipelines:
<pre>
public class DataServerInitializer extends [@link ChannelInitializer]&lt;[@link Channel]&gt; {

    private static final DataServerHandler SHARED = new DataServerHandler();

    @Override
    public void initChannel([@link Channel] channel) {
        channel.pipeline().addLast("handler", SHARED);
    }
}</pre>

The [@code @Sharable] annotation
In the example above which used an [@link AttributeKey],
you might have noticed the [@code @Sharable] annotation.
If a `@link ChannelHandler` is annotated with the `@Sharable` annotation, it means you can create an instance of the handler just once and add it to one or more `@link ChannelPipeline`s multiple times without a race condition.

If this annotation is not specified, you have to create a new handler instance every time you add it to a pipeline because it has unshared state such as member variables.

This annotation is provided for documentation purpose, just like the JCIP annotations.

Please refer to the `@link ChannelHandler`, and `@link ChannelPipeline` to find out more about inbound and outbound operations, what fundamental differences they have, how they flow in a pipeline, and how to handle the operation in your application.

Additional resources worth reading

Please refer to the `@link ChannelHandler`, and `@link ChannelPipeline` to find out more about inbound and outbound operations, what fundamental differences they have, how they flow in a pipeline, and how to handle the operation in your application.

Found in path(s):

No license file was found, but licenses were detected in source scan.
* The encoding and decoding algorithm in this class has been derived from
* Robert Harder's Public Domain Base64 Encoder/Decoder.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/base64/Base64.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/ocsp/OcspRequestBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/ocsp/OcspClientExample.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/ocsp/OcspServerExample.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/ocsp/Digester.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/example/ocsp/OcspUtils.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/*
* Written by Robert Harder and released to the public domain, as explained at
* https://creativecommons.org/licenses/publicdomain
*/
/**
* Enumeration of supported Base64 dialects.
* <p>
* The internal lookup tables in this class has been derived from
* <a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain
* Base64 Encoder/Decoder</a>.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/base64/Base64Dialect.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version
* 2.0 (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/flow/FlowControlHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/ByteProcessorUtils.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/util/internal/ResourcesUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/redis/InlineCommandRedisMessage.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version
* 2.0 (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
jar/io/netty/handler/codec/http/cors/CorsConfigBuilder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/internal/tcnative/CertificateRequestedCallback.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/internal/tcnative/SessionTicketKey.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/util/ByteProcessor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/UniformStreamByteDistributor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/DecoratingHttp2ConnectionDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/StreamByteDistributor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/AbstractInboundHttp2ToHttpAdapterBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-
  jar/io/netty/handler/codec/http2/StreamBufferingEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/WeightedFairQueueByteDistributor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2PromisedRequestVerifier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/DecoratingHttp2ConnectionEncoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/ValueConverter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/InboundHttp2ToHttpAdapterBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamVisitor.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/CharSequenceValueConverter.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/CoalescingBufferQueue.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/MathUtil.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/CleanerJava9.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/concurrent/FastThreadLocalRunnable.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/internal/Cleaner.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2020 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/internal/tcnative/SSLSessionCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/internal/tcnative/SSLSession.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http/websocketx/WebSocketChunkedInput.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/concurrent/PromiseCombiner.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/PemEncoded.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/IntSupplier.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/package-info.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2GoAwayFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/util/concurrent/RejectedExecutionHandlers.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/kqueue/KQueueRecvByteAllocatorHandle.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/kqueue/KQueueChannelOption.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/flush/FlushConsolidationHandler.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/channel/unix/PeerCredentials.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ReferenceCountedOpenSslEngine.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2StreamFrameToHttpObjectCodec.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ssl/TrustManagerFactory.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/ssl/ssl/X509TrustManagerWrapper.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2ResetFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/http2/Http2ResetFrameToHttpObjectCodec.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpUtils.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/DefaultSmtpRequest.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpObjectCodec.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpObjectCodecBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpObjectCodecBuilderImpl.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpObjectCodecImpl.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStatistics.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpStatsUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStats.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStatsBuilder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStatsBuilderImpl.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStatsImpl.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/handler/codec/smtp/SmtpServerStatsUtil.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources-jar/io/netty/buffer/PooledSlicedByteBuf.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/PreferHeapByteBufAllocator.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2PingFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/resolver/dns/DnsCache.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/AbstractPooledDerivedByteBuf.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/kqueue/KQueueDatagramChannel.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/DefaultHttp2WindowUpdateFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/kqueue/KQueueEventLoopGroup.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/channel/epoll/LinuxSocket.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/smtp/SmtpResponseDecoder.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/dns/AbstractDnsOptPseudoRrRecord.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/DefaultHttp2DataFrame.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/ssl/ReferenceCountedOpenSslClientContext.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/PooledDuplicatedByteBuf.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/buffer/AbstractUnpooledSlicedByteBuf.java
* /opt/cola/permits/1119092601_1614145555.72/0/netty-all-4-1-56-final-sources.jar/io/netty/handler/codec/http2/Http2Frame.java

Open Source Used In DNAC Platform Arica 11546
1.2450 jarschivelib 1.0.0

1.2450.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2451 batik-transcoder 1.13

1.2451.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/svg2svg/OutputManager.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/FloatKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/AbstractTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/TranscoderException.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/print/PrintTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/WMFConstants.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/IntegerKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-
Open Source Used In DNAC Platform Arica  11553

jar/org/apache/batik/transcoder/svg2svg/PrettyPrinter.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/AbstractTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/Rectangle2DKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/ErrorHandler.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/PaintKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/WMFRecordStore.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/Transcoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/AbstractWMFReader.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/image/resources/Messages.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/svg2svg/SVGTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/StringKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/image/TIFFTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/TranscoderSupport.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/LengthKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/RecordStore.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/MetaRecord.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/image/JPEGTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/image/PNGTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/BooleanKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/AbstractWMFPainter.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/WMFFont.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/DefaultErrorHandler.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-
jar/org/apache/batik/transcoder/XMLAbstractTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/TranscoderOutput.java
* /opt/ cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/WMFTranscoder.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/WMFPainter.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/TranscodingHints.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/keys/DOMImplementationKey.java
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/TranscoderInput.java

No license file was found, but licenses were detected in source scan.

# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/image/resources/Messages.properties
No license file was found, but licenses were detected in source scan.

Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1146878592_1616810445.17/0/batik-transcoder-1-13-sources-jar/org/apache/batik/transcoder/wmf/tosvg/WMFUtilities.java

1.2452 jsonfile 2.4.0 2.4.0

1.2452.1 Available under license :
(The MIT License)

Copyright (c) 2012-2015, JP Richardson <jprichardson@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
1.2453 anymatch 3.1.1 3.1.1

1.2453.1 Available under license:
The ISC License

Copyright (c) 2019 Elan Shanker, Paul Miller (https://paulmillr.com)

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2454 parquet-column 1.10.0

1.2454.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

------------------------------------------------------------------------

This product depends on fastutil and includes it in this binary artifact.
Fastutil provides type-specific collection implementations.

Copyright: 2002-2014 Sebastiano Vigna
Home page: http://fasutil.di.unimi.it/
License: http://www.apache.org/licenses/LICENSE-2.0.html

1.2455 jackson-annotations 2.4.0
1.2455.1 Available under license :
This copy of Jackson JSON processor annotations is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.2456 @babel/plugin-proposal-class-static-block 7.16.0
1.2456.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2457 babel-code-frame 6.26.0
1.2457.1 Available under license :
MIT

1.2458 is-equal-shallow 0.1.3
1.2458.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2459 node-fetch 1.7.3 1.7.3

1.2459.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 David Frank

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2460 tmp 0.0.33 0.0.33

1.2460.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 KARASZI Istvn

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, 
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE 
SOFTWARE.

1.2461 natural-compare 1.4.0
1.2461.1 Available under license:
No license file was found, but licenses were detected in source scan.

[The MIT License](http://lauri.rooden.ee/mit-license.txt)

Found in path(s):
* /opt/cola/permits/1237623697_1638828154.57/0/natural-compare-1.4.0-tgz/package/README.md
No license file was found, but licenses were detected in source scan.

/*
 * @version    1.4.0
 * @date       2015-10-26
 * @stability 3 - Stable
 * @author     Lauri Rooden (https://github.com/litejs/natural-compare-lite)
 * @license    MIT License
 */

Found in path(s):
* /opt/cola/permits/1237623697_1638828154.57/0/natural-compare-1.4.0-tgz/package/index.js

1.2462 readdirp 2.2.1
1.2462.1 Available under license:
MIT

1.2463 drill-memory-base 1.19.0
1.2463.1 Available under license:

Drill : Exec : Memory : Base
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2464 jackson-dataformat-yaml 2.10.0

1.2464.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.2465 resolve-from 5.0.0

1.2465.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2466 commons-configuration 2.1.1

1.2466.1 Available under license :

Apache Commons Configuration
Copyright 2001-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2467 ext 1.4.0

1.2467.1 Available under license:

ISC License

Copyright (c) 2011-2019, Mariusz Nowak, @medikoo, medikoo.com

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2468 flux 3.1.3 3.1.3
1.2468.1 Available under license :
    BSD License
    
    For Flux software
    
    Copyright (c) 2014-present, Facebook, Inc. All rights reserved.
    
    Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
    
    * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
    
    * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
    
    * Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
    
    THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2469 normalize-path 3.0.0
1.2469.1 Available under license :
    The MIT License (MIT)
    
    Copyright (c) 2014-2018, Jon Schlinkert.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2470 parquet-hadoop 1.12.0

1.2470.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product includes code from Twitter's ElephantBird project.

* parquet-hadoop's UnmaterializableRecordCounter.java includes code from ElephantBird's LzoRecordReader.java

Copyright: 2012-2014 Twitter
Home page: https://github.com/twitter/elephant-bird
License: http://www.apache.org/licenses/LICENSE-2.0

1.2471 node-fetch-h 2.3.0 2.3.0

1.2471.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 David Frank

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2472 is-boolean-object 1.1.0

1.2472.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2473 sparsebitset 1.2

1.2473.1 Available under license:

No license file was found, but licenses were detected in source scan.

```xml
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.zaxxer</groupId>
  <artifactId>SparseBitSet</artifactId>
  <version>1.2</version>
  <packaging>jar</packaging>

  <name>SparseBitSet</name>
  <description>An efficient sparse bitset implementation for Java</description>
  <url>https://github.com/brettwooldridge/SparseBitSet</url>

  <organization>
    <name>Zaxxer.com</name>
    <url>https://github.com/brettwooldridge/SparseBitSet</url>
  </organization>
</project>
```
1.2474 lodash.uniq 4.5.0 4.5.0

1.2474.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2475 hbase-protocol-shaded 2.4.2

1.2475.1 Available under license :

Apache HBase - Shaded Protocol
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2476 babel-plugin-transform-es2015-function-name 6.24.1

1.2476.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>

Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical ReportsCopyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2477 pluralize 1.1.6 1.1.6
1.2477.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2013 Blake Embrey (hello@blakeembrey.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2478 is-glob 4.0.1 4.0.1
1.2478.1 Available under license :
   The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2479 remove-trailing-separator 1.1.0 1.1.0
1.2479.1 Available under license:

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2480 kind-of 5.1.0
1.2480.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2481 hoist-non-react-statics 2.5.5 2.5.5

1.2481.1 Available under license :
Software License Agreement (BSD License)
========================================

Copyright (c) 2015, Yahoo! Inc. All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Yahoo! Inc. nor the names of YUI's contributors may be used to endorse or promote products derived from this software without specific prior written permission of Yahoo! Inc.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.2482 de.huxhorn.lilith.data.logging 0.9.44

1.2482.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser
General Public License, and the “GNU GPL” refers to version 3 of the
GNU General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

/*
 * Lilith - a log event viewer.
 * Copyright (C) 2007-2011 Joern Huxhorn
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU Lesser General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/*
 * Copyright 2007-2011 Joern Huxhorn
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* * http://www.apache.org/licenses/LICENSE-2.0
* * Unless required by applicable law or agreed to in writing, software
* * distributed under the License is distributed on an "AS IS" BASIS,
* * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* * See the License for the specific language governing permissions and
* * limitations under the License.
*/

1.2483 esri-geometry-api 2.2.2
1.2483.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/CombineOperator.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2017 Esri
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

e-mail: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCentroid2DLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorImportFromJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCMultiPoint.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JSONUtils.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/AttributeStreamOfFloat.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/AttributeStreamOfDbl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JsonParserReader.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCMultiCurve.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorImportFromGeoJson.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorFactoryLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCMultiSurface.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorImportFromJson.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCMultiSurface.java

Open Source Used In DNAC Platform Arica 11600
jar/com/esri/core/geometry/AttributeStreamOfInt8.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JsonReader.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/AttributeStreamOfInt64.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/AttributeStreamOfInt16.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCCurve.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCentroid2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCLinearRing.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCLinearRing.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCLinearRing2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCGeometryCollection.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OGCSurface.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JsonGeometryException.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorImportFromJsonCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/SimpleJsonReaderCursor.java

No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2015 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/ShapeImportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorExportToWktLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/SweepComparator.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorIntersectionLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/TopologicalOperations.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorWithinLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorConvexHull.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/IntervalTreeImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorOffset.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/Point3D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorEqualsLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorUnionCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/PtSrIzr.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/MultiVertexGeometryImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorShapePreservingDensifyLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorBufferLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/PlaneSweepCrackerHelper.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OGCStructure.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorSymmetricDifference.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorBuffer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorDensifyByLengthLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OperatorImportFromESRIShape.java

Open Source Used In DNAC Platform Arica 11602
Open Source Used In DNAC Platform Arica 11603

* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorIntersects.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorGeodeticLength.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/UserCancelException.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorDisjoint.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/SpatialReferenceImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Boundary.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/ClassicSort.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/MultiPath.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeometryException.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Wkt.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorGeneralizeCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorExportToWkt.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GenericGeometrySerializer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/MultiPathImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorDisjointLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimplifyCursorOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorCrosses.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorSimpli
Open Source Used In DNAC Platform Arica 11604

* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorConvexHullCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSimplifyLocalHelper.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorRelateLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/Treap.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorProximity2DLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorEquals.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/ConstructorOffset.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorExportToWkb.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/VertexDescriptionHash.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorGeodeticAreaLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/SegmentIterator.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSimplifyCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/MultiVertexGeometry.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorCrossesLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/GeoJsonCrsTables.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorBoundaryLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/ObjectCacheTable.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/Clusterer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/WkbByteOrder.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorDistance.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/JsonStringWriter.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/NonSimpleResult.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Proximity2DResultComparator.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/SegmentBuffer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorGeneralize.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorProjectLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Simplificator.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorExportToJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Envelope1D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/RingOrientationFixer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeoDist.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperateExportToJsonCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JsonCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorClipCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorDifferenceLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/MapOGCStructure.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorExportToWkbLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorDistanceLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorOverlaps.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/GeometrySerializer.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorGeodeticLengthLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorDensifyByLength.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/NumberUtils.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorGeodeticArea.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSimplify.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/RelationalOperations.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/RasterizedGeometry2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/ByteBufferCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorContains.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/SweepMonkierComparator.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorRelate.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/GeometryAccelerators.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/ShapeExportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSimpleRelation.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSimplifyOGC.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/PolygonUtils.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorExportToESRIShapeLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/Clipper.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/OperatorSymmetricDifferenceCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources.jar/com/esri/core/geometry/Transformation3D.java
Open Source Used In DNAC Platform Arica  11608
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/MathUtils.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/WktImportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Transformation2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/WktParser.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorBoundary.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Wkid.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeoJsonExportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorDensifyByLengthCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/WkbImportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/TopoGraph.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorTouchesLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/RelationalOperationsMatrix.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorTouches.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/SegmentIntersector.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/WktExportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorOffsetLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorClip.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/MultiPoint.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeometryCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorIntersectionCursor.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/SegmentIteratorImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeoJsonImportFlags.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/CrackAndCluster.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2013 Esri

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

email: contracts@esri.com

*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/envelope3d-java-gz/Envelope3D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/point3d-java-gz/Point3D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/RasterizedGeometry2DImpl.java

No license file was found, but licenses were detected in source scan.

*/

Copyright 1995-2017 Esri

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/
/*
COPYRIGHT 1995-2017 ESRI

TRADE SECRETS: ESRI PROPRIETARY AND CONFIDENTIAL
Unpublished material - all rights reserved under the
Copyright Laws of the United States.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/JsonReaderCursor.java
No license file was found, but licenses were detected in source scan.

*/
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorExportToGeoJson.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 1995-2018 Esri

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373

email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/OperatorExportToGeoJsonLocal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Polygon.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/StridedIndexTypeCollection.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/QuadTreeImpl.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/Polyline.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/ogc/OGCPolygon.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/GeometryEngine.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/ogc/OGCConcreteGeometryCollection.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/ogc/OGCMultiPolygon.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/QuadTree.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-jar/com/esri/core/geometry/ogc/OGCPoint.java
jar/com/esri/core/geometry/SpatialReference.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/Envelope.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/ogc/OGCLineString.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OGCGeometry.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/Geometry.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/Point2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/OGCStructureInternal.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/ogc/OGCPoint.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/AttributeStreamOfInt32.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/MapGeometry.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/SizeOf.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/Envelope2D.java
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/ogc/OGCMultiLineString.java

No license file was found, but licenses were detected in source scan.

/*
Copyright 2013-2015 Esri
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

For additional information, contact:
Environmental Systems Research Institute, Inc.
Attn: Contracts Dept
380 New York Street
Redlands, California, USA 92373
email: contracts@esri.com
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092327381_1600729703.66/0/esri-geometry-api-2-2-2-sources-
jar/com/esri/core/geometry/SimpleRasterizer.java

1.2484 lodash 4.17.21 4.17.21
1.2484.1 Available under license:
Software License Agreement (BSD License)

Copyright (c) 2007, Parakey Inc.
All rights reserved.

Redistribution and use of this software in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above
copyright notice, this list of conditions and the
following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other
materials provided with the distribution.

* Neither the name of Parakey Inc. nor the names of its
contributors may be used to endorse or promote products
derived from this software without specific prior
written permission of Parakey Inc.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2485 colors.js 1.4.0 1.4.0

1.2485.1 Available under license:

MIT License

Original Library
- Copyright (c) Marak Squires

Additional Functionality
- Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2486 remove-trailing-separator 1.1.0

1.2486.1 Available under license:

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2487 zlib 1.2.8

1.2487.1 Available under license:

/* zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jlo@gzip.org          madler@alumni.caltech.edu

*/

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.2488 listenablefuture 1.0

**1.2488.1 Available under license:**

No license file was found, but licenses were detected in source scan.

```java
/*
 * Copyright (C) 2007 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258121121_1643033053.9/0/listenablefuture-1-0-sources-jar/com/google/common/util/concurrent/ListenableFuture.java
```

### 1.2489 restore-cursor 2.0.0 2.0.0

**1.2489.1 Available under license:**

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2490 @babel/plugin-bugfix-v8-spread-parameters-in-optional-chaining 7.16.0
1.2490.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2491 path-root-regex 0.1.2 0.1.2
1.2491.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2492 imurmurhash 0.1.4
1.2492.1 Available under license:
No license file was found, but licenses were detected in source scan.

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
subject to the following conditions:
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

Found in path(s):
* /opt/cola/permits/1110635849_1607035374.31/0/imurmurhash-0-1-4-4-tgz/package/README.md

1.2493 htrace-core 3.2.0-incubating
1.2493.1 Available under license:
htrace-core
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
1.2494 oas-schema-walker 1.1.5 1.1.5

1.2494.1 Available under license:
BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2495 use 3.1.1 3.1.1

1.2495.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2496 inquirer 6.2.2 6.2.2
1.2496.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2497 @babel/types 7.16.0

1.2497.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2498 is-binary-path 2.1.0

1.2498.1 Available under license:
MIT License

Copyright (c) 2019 Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com), Paul Miller (https://paulmillr.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
1.2499 jnr-posix 3.0.44

1.2499.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
to your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

***** BEGIN LICENSE BLOCK *****
* Version: CPL 1.0/GPL 2.0/LGPL 2.1
*
***** END LICENSE BLOCK *****
1.2500 read 1.0.7 1.0.7

1.2500.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2501 utils-merge 1.0.1

1.2501.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013-2017 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2502 is-primitive 2.0.0

1.2502.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2503 @babel/generator 7.16.0
1.2503.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2504 ono 4.0.11 4.0.11
1.2504.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 James Messinger

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software,
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2505 callsite 1.0.0 1.0.0
1.2505.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2506 xregexp 4.0.0 4.0.0
1.2506.1 Available under license:
The MIT License

Copyright (c) 2007-2017 Steven Levithan <http://xregexp.com/>.Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2507 range-parser 1.2.0 1.2.0

1.2507.1 Available under license :

(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015-2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, publish, distribute,
sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2508 guava 11.0.2

1.2508.1 Available under license:

Apache License

Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement You may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

vendor: https://code.google.com/p/guava-libraries/
license: http://www.apache.org/licenses/LICENSE-2.0
source: https://code.google.com/p/guava-libraries/source/browse/?name=v11.0.2

1.2509 is-ip 1.0.0
1.2509.1 Available under license :
    MIT

1.2510 object.getownpropertydescriptors
2.1.2
1.2510.1 Available under license :
    The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2511 is-buffer 2.0.5
1.2511.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2512 @babel/plugin-bugfix-safari-id-destructuring-collision-in-function-expression 7.16.2
1.2512.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2513 optionator 0.8.2
1.2513.1 Available under license :
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2514 passport-local 1.0.0 1.0.0
1.2514.1 Available under license :
The MIT License (MIT)

Copyright (c) 2011-2014 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2515 httpcore5 5.0.2
1.2515.1 Available under license :

Apache HttpComponents Core HTTP/1.1
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2516 netty-handler 4.0.48.Final

1.2516.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-jar/io/netty/handler/traffic/TrafficCounter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// Try the OpenJDK's proprietary implementation.

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-jar/io/netty/handler/ssl/util/SelfSignedCertificate.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
* /

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/SslCompletionEvent.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/Ocsp/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/SslCloseCompletionEvent.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/OptionalSslHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/ConscryptAlpnSslEngine.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/ocsp/OcspClientHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/util/BouncyCastleSelfSignedCertGenerator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/CipherSuiteFilter.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/JdkDefaultApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/OpenSslApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/Java7SslParametersUtils.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/OpenSslServerSessionContext.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ipfilter/IpSubnetFilterRule.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/IdentityCipherSuiteFilter.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/ApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JettyAlpnSslEngine.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/util/InsecureTrustManagerFactory.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/ApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkSslEngine.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/OpenSsl.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/OpenSslServerContext.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/util/ThreadLocalInsecureRandom.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ipfilter/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ipfilter/IpFilterRule.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkSslEngine.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/CipherSuiteConverter.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkSslSession.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkSslServerContext.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ipfilter/UniquelIpFilter.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/OpenSslX509Certificate.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkBaseApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ipfilter/RuleBasedIpFilter.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/util/FingerprintTrustManagerFactory.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/OpenSslEngineMap.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/ApplicationProtocolConfig.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/JdkAlpnApplicationProtocolNegotiator.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 The Netty Project
 * The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/traffic/AbstractTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
jar/io/netty/handler/ssl/SslHandshakeCompletionEvent.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE
2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*/
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

Found in path(s):
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ DelegatingSslContext.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ PemX509Certificate.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ flush/ package-info.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ util/ X509TrustManagerWrapper.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ Java8SslUtils.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ flush/ FlushConsolidationHandler.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ PemEncoded.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ PemValue.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ OpenSslKeyMaterialManager.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ OpenSslExtendedKeyMaterialManager.java
/ opt/ cola/ permits/ 1202446677_ 1631568413. 44/ 0/ netty- handler- 4- 0- 48- final-sources- jar/ io/ netty/ handler/ ssl/ PemPrivateKey.java

No license file was found, but licenses were detected in source scan.

/*

Copyright 2012 The Netty Project

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/traffic/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/traffic/GlobalTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/traffic/GlobalChannelTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/ssl/NotSslRecordException.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/logging/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/ChunkedInput.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/IdleState.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/ReadTimeoutException.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/ChunkedFile.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/logging/LoggingHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/ReadTimeoutHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/WriteTimeoutHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/package-info.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/IdleStateHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/ChunkedWriteHandler.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/ChunkedNioFile.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/stream/ChunkedNioStream.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/logging/LogLevel.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
  jar/io/netty/handler/timeout/TimeoutException.java
* /opt/cola/permits/1202446677_1631568413.44/0/netty-handler-4-0-48-final-sources-
1.2517 es6-iterator 2.0.3

1.2517.1 Available under license:

The MIT License (MIT)

Copyright (C) 2013-2017 Mariusz Nowak (www.medikoo.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2518 codemirror 5.59.2 5.59.2

1.2518.1 Available under license:

MIT License

Copyright (C) 2017 by Marijn Haverbeke <marijnh@gmail.com> and others

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2519 generate-object-property 1.2.0
1.2519.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2520 batik-slideshow 1.13
1.2520.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
1.2521.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2522 is-binary-path 1.0.1

1.2522.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.

1.2523 escape-goat 2.1.1
1.2523.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2524 regenerate 1.4.0 1.4.0
1.2524.1 Available under license:
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.2525 perfmark-api 0.19.0

1.2525.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

“License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

“Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

“Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, “control” means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

“You” (or “Your”) shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   You must give any other recipients of the Work or Derivative Works a copy of this License; and

   You must cause any modified files to carry prominent notices stating that You changed the files; and

   You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2526 color-name 1.1.1
1.2526.1 Available under license:
   The MIT License (MIT)
   Copyright (c) 2015 Dmitry Ivanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2527 log4j-over-slf4j 1.7.26

1.2528 mail 1.4.7
1.2528.1 Available under license:
   COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

   Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

   (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION
OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item,. as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a
consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-fee redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2529 to-regex 3.0.2 3.0.2

1.2529.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2530 jetty-http 9.4.41.v20210516
1.2530.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.
"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:
a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties
      and conditions, express and implied, including warranties or
      conditions of title and non-infringement, and implied warranties or
      conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for
       damages, including direct, indirect, special, incidental and
       consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are
        offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such
       Contributor, and informs licensees how to obtain it in a reasonable
       manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted
under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
==================================================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN
module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.2531 babel-plugin-transform-es2015-modules-umd 6.24.1

1.2531.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2532 string-width 3.1.0 3.1.0

1.2532.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2533 netty-handler-proxy 4.1.42.Final

1.2533.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/package-info.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/Socks4ProxyHandler.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/HttpProxyHandler.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/ProxyConnectionEvent.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/ProxyConnectException.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/ProxyHandler.java
* /opt/cola/permits/1258272209_1643050042.71/0/netty-handler-proxy-4-1-42-final-sources-
  jar/io/netty/handler/proxy/Socks5ProxyHandler.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

1.2534 core-js 2.6.12 2.6.12
1.2534.1 Available under license:
Copyright (c) 2014-2020 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2535 regjsparsers 0.1.5
1.2535.1 Available under license:
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2536 ansi-styles 3.2.1 3.2.1
1.2536.1 Available under license :
MIT License
Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2537 release-zalgo 1.0.0
1.2537.1 Available under license :
ISC License (ISC)
Copyright (c) 2017, Mark Wubben <mark@novemberborn.net> (novemberborn.net)

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2538 @babel/plugin-transform-block-scoping 7.16.0
1.2538.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2539 async 1.5.2 1.5.2
1.2539.1 Available under license:
Copyright (c) 2010-2014 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2540 commons-math 2.1

1.2540.1 Available under license:
Apache Commons Math
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

===============================================================================
The LinearConstraint, LinearObjectiveFunction, LinearOptimizer, Relationship, SimplexSolver and SimplexTableau classes in package org.apache.commons.math.optimization.linear include software developed by Benjamin McCann (http://www.benmccann.com) and distributed with the following copyright: Copyright 2009 Google Inc.
===============================================================================

This product includes software developed by the
University of Chicago, as Operator of Argonne National Laboratory.
The LevenbergMarquardtOptimizer class in package
org.apache.commons.math.optimization.general includes software translated from the lmder, lmpar and qrsolv Fortran routines from the Minpack package
Minpack Copyright Notice (1999) University of Chicago. All rights reserved
===============================================================================

The GraggBulirschStoerIntegrator class in package
org.apache.commons.math.ode.nonstiff includes software translated from the odex Fortran routine developed by E. Hairer and G. Wanner.
Original source copyright:
Copyright (c) 2004, Ernst Hairer
===============================================================================

The EigenDecompositionImpl class in package org.apache.commons.math.linear includes software translated from some LAPACK Fortran routines. Original source copyright: Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

The MersenneTwister class in package org.apache.commons.math.random includes software translated from the 2002-01-26 version of the Mersenne-Twister generator written in C by Makoto Matsumoto and Takuji Nishimura. Original source copyright: Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved

The complete text of licenses and disclaimers associated with the the original sources enumerated above at the time of code translation are in the LICENSE.txt file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS MATH DERIVATIVE WORKS:

The Apache commons-math library includes a number of subcomponents whose implementation is derived from original sources written in C or Fortran. License terms of the original sources are reproduced below.

===============================================================================
For the lmder, lmpar and qrsolv Fortran routine from minpack and translated in the LevenbergMarquardtOptimizer class in package org.apache.commons.math.optimization.general
Original source copyright and license statement:

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

   "This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

   Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WarnED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

===============================================================================

Open Source Used In DNAC Platform Arica 11709
Copyright and license statement for the odex Fortran routine developed by E. Hairer and G. Wanner and translated in GraggBulirschStoerIntegrator class in package org.apache.commons.math.ode.nonstiff:

Copyright (c) 2004, Ernst Hairer

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

Copyright and license statement for the original lapack fortran routines translated in EigenDecompositionImpl class in package org.apache.commons.math.linear:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

$COPYRIGHT$

Additional copyrights may follow

$HEDERS$

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.

- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===============================================================================

Copyright and license statement for the original Mersenne twister C routines translated in MersenneTwister class in package org.apache.commons.math.random:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2541 type-check 0.3.2
1.2541.1 Available under license:
Copyright (c) George Zahariev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2542 babel-helper-builder-binary-assignment-operator-visitor 6.24.1
1.2542.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2543 annotations 16.0.3

1.2543.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2006 Sascha Weinreuter
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/PrintFormat.java
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/Identifier.java
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/Subst.java
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/Pattern.java
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/Language.java
*/opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-
jar/org/intellij/lang/annotations/RegExp.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2000-2016 JetBrains s.r.o.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-jar/org/jetbrains/annotations/Nls.java
* /opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-jar/org/intellij/lang/annotations/Flow.java
* /opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-jar/org/jetbrains/annotations/TestOnly.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2000-2014 JetBrains s.r.o.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-jar/org/jetbrains/annotations/Nullable.java
* /opt/ws_local/PERMITS_SQL/1066932998_1593721013.75/0/annotations-16-0-3-sources-jar/org/intellij/lang/annotations/MagicConstant.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2000-2009 JetBrains s.r.o.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
1.2544 batik-parser 1.13

1.2544.1 Available under license:

Apache Batik

Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This software contains code from the World Wide Web Consortium (W3C) for the
Document Object Model API (DOM API) and SVG Document Type Definition (DTD).

This software contains code from the International Organisation for
Standardization for the definition of character entities used in the software’s
documentation.

This product includes images from the Tango Desktop Project
(http://tango.freedesktop.org/).

This product includes images from the Pasodoble Icon Theme

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2545 mitt 1.1.3 1.1.3

1.2545.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017 Jason Miller

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2546 js-string-escape 1.0.1 1.0.1

1.2546.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013 Jo Liss

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2547 babel-messages 6.23.0 6.23.0
1.2547.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2548 opentest4j 1.2.0

1.2548.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2018 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
1.2549 babel-helper-call-delegate 6.24.1

1.2549.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*. 
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2550 netty-resolver 4.1.42.Final
1.2550.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4-1-42-final-sources-3-jar/io/netty/resolver/RoundRobinInetAddressResolver.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0

~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/META-INF/maven/io.netty/netty-resolver/pom.xml
No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/NameResolver.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/AddressResolverGroup.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/package-info.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/NoopAddressResolver.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/SimpleNameResolver.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4.1.42-final-sources-3-jar/io/netty/resolver/NoopAddressResolverGroup.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4-1-42-final-sources-3.jar/io/netty/resolver/HostsFileEntries.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4-1-42-final-sources-3.jar/io/netty/resolver/ResolvedAddressTypes.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4-1-42-final-sources-3.jar/io/netty/resolver/InetNameResolver.java
* /opt/cola/permits/1144318097_1615901354.89/0/netty-resolver-4-1-42-final-sources-3.jar/io/netty/resolver/DefaultNameResolver.java
1.2551 aws-java-sdk-s3 1.11.704

1.2551.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/AmazonS3EncryptionClient.java
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/MultipartUploadCryptoContext.java
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/ContentCryptoMaterial.java
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/S3KeyWrapScheme.java
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/S3CryptoModuleAE.java
* /opt/cola/permits/1137154767_1615901354.89/0/aws-java-sdk-s3-1-11-704-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015-2020 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 *    http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketAccelerateConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetBucketReplicationConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketLoggingConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketTaggingConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketVersioningConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ListNextBatchOfVersionsRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketLifecycleConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/DownloadCallable.java
/
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2020 Amazon Technologies, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/PresignedUrlRetryableDownloadTaskImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at:
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/DownloadS3ObjectCallable.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/PreparedDownloadTask.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ScanRange.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3AccessPointBuilder.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/ObjectCannedAclProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon Technologies, Inc.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/BucketLifecycleConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/ObjectRestoreResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ResponseHeaderOverrides.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetBucketNotificationConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/BucketCrossOriginConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/CORSRule.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ResponseHeaderHandlerChain.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/ServerSideEncryptionResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/ResponseHeaderHandlerChain.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/internal/TransferMonitor.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 * http://aws.amazon.com/apache2.0
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/replication/ReplicationFilterPredicate.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/metrics/MetricsConfiguration.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/GetBucketInventoryConfigurationResult.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/GetBucketAnalyticsConfigurationRequest.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/replication/ReplicationPredicateVisitor.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/ListBucketAnalyticsConfigurationsResult.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/GetBucketMetricsConfigurationRequest.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/ListBucketInventoryConfigurationsRequest.java
jar/com/amazonaws/services/s3/model/DeleteBucketInventoryConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ListBucketMetricsConfigurationsResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryOptionalField.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ListBucketAnalyticsConfigurationsRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleAndOperator.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysis.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleTagPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventorySchedule.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryIncludedObjectVersions.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysisSchemaVersion.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/metrics/MetricsTagPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/AnalyticsS3BucketDestination.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/metrics/MetricsFilter.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/replication/ReplicationAndOperator.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/DeleteBucketAnalyticsConfigurationResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryOptionalField.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleAndOperator.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysis.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleTagPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryOptionalField.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleAndOperator.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysis.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/lifecycle/LifecycleTagPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/StorageClassAnalysisSchemaVersion.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/metrics/MetricsTagPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/analytics/AnalyticsFilterPredicate.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/InventoryConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Grantee.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/XmlResponsesSaxParser.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/S3StringResponseHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/BucketLoggingConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/AccessControlList.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/BucketPolicy.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Grant.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/EmailAddressGrantee.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/BucketTaggingConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/RestUtils.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/RepeatableInputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/BucketNotificationConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/RequestPaymentConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/CanonicalGrantee.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/TagSet.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/ServiceUtils.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Bucket.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016. Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/WebsiteConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/internal/SkipMd5CheckStrategy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/KMSEncryptionMaterials.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License"). You may not
* use this file except in compliance with the License. A copy of the License is
* located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed on
* an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/waiters/AmazonS3Waiters.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/inventory/InventoryEncryption.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/DeleteMarkerReplicationStatus.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/SseKmsEncryptedObjects.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/ExistingObjectReplication.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-

Open Source Used In DNAC Platform Arica 11743
jar/com/amazonaws/services/s3/model/inventory/ServerSideEncryptionS3.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SourceSelectionCriteria.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetBucketEncryptionResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/inventory/ServerSideEncryptionKMS.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/DeleteBucketEncryptionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetBucketEncryptionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetBucketEncryptionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/DeleteMarkerReplication.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ExistingObjectReplicationStatus.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ServerSideEncryptionByDefault.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License"). You may not
 * use this file except in compliance with the License. A copy of the License is
 * located at
 * 
 * http://aws.amazon.com/apache2.0
 * 
 * or in the "license" file accompanying this file. This file is distributed on
 * an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */
Open Source Used In DNAC Platform Arica 11745

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/waiters/HeadBucketFunction.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/waiters/HeadObjectFunction.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2013-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
* /

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/AbstractRepeatableCipherInputStream.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* 
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* 
* http://aws.amazon.com/apache2.0
* 
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
* /
*/
/**
* Synchronous client for accessing Amazon S3.
* *
* <h1>Appendix: Amazon S3 client-side encryption meta information</h1>
This appendix summarizes the current crypto related meta information associated with an S3 object encrypted using SDK client-side encryption. In SDK's terminology, these meta information are classified as "user metadata". In general, the meta information is just a map of key/value pairs. Physically they are persisted either as metadata of the encrypted S3 object, or as a separate S3 object called an "instruction file" (which is just a JSON file in plaintext.) For all references to the Cipher Algorithm Names, Modes, and Padding, such as <code>"AES/CBC/PKCS5Padding"</code>, please refer to Oracle's <a href="http://docs.oracle.com/javase/7/docs/technotes/guides/security/StandardNames.html">Java Cryptography Architecture Standard Algorithm Name Documentation</a> for more details.

Note when a key/value pair is stored as "user metadata" as part of the S3 metadata, the SDK always adds an additional prefix of <b>x-amz-meta-</b> to the key name. In contrast, the SDK doesn't add any prefix when the key names are stored in an instruction file. (The reason has to do with how the http headers are handled.)

For example, <b>x-amz-key</b> would get stored as <b>x-amz-meta-x-amz-key</b> in S3 metadata. In contrast, <b>x-amz-key</b> would simply be stored as is (ie without the prefix) in an instruction file. All the keys described below are considered "user metadata" by the SDK.

For non-KMS client side encryption, this v1 crypto meta information is used to support the "Encryption Only" crypto mode (which involves the use of <code>AES/CBC/PKCS5Padding</code> for content encryption, and <code>AES/ECB</code> for encrypting the one-time randomly generated data key). For KMS client-side encryption, the v2 crypto meta information is used.

<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-key</td>
<td>Content encrypting key (cek) in encrypted form, base64 encoded. The cek is randomly generated per S3 object, and is always an AES 256-bit key. The corresponding cipher is always &quot;AES/CBC/PKCS5Padding&quot;.</td>
</tr>
</tbody>
</table>
<table border="1">
<table>
<thead>
<tr>
<th>key</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-key-v2</td>
<td>CEK in key wrapped form. This is necessary so that the S3 encryption client doesn't recognize the v2 format will not mistakenly decrypt S3 object encrypted in v2 format.</td>
</tr>
<tr>
<td>x-amz-iv</td>
<td>Randomly generated IV (per S3 object), base64 encoded. (Same as v1.)</td>
</tr>
<tr>
<td>x-amz-matdesc</td>
<td>Customer provided material description in JSON format. (Same as v1). Used to identify the client-side master key.</td>
</tr>
<tr>
<td></td>
<td>For KMS client side encryption, the KMS Customer Master Key ID is stored as part of the material description, &lt;b&gt;x-amz-matdesc&lt;/b&gt;, under the key-name &lt;code&gt;&quot;kms_cmk_id&quot;&lt;/code&gt;.</td>
</tr>
</tbody>
</table>
</table>
* </ul>
* </td>
* </tr>
* <tr>
* <td><b>x-amz-unencrypted-content-length</b></td>
* <td>Unencrypted content length. (optional but should be specified whenever possible. Same as v1.)</td>
* </tr>
* <tr>
* <td><b>x-amz-wrap-alg</b></td>
* <td>Key wrapping algorithm used.
* <ul>
* <li>Supported values:
* <code>"AESWrap", "RSA/ECB/OAEPWithSHA-256AndMGF1Padding", "kms"</code></li>
* <li>No standard key wrapping is used if this meta information is absent</li>
* <li>Always set to <b>"kms"</b> if KMS is used for client-side encryption</li>
* </ul>
* </tr>
* <tr>
* <td><b>x-amz-cek-alg</b></td>
* <td>Content encryption algorithm used.
* <ul>
* <li>Supported values:
* <code>"AES/GCM/NoPadding", "AES/CBC/PKCS5Padding"</code></li>
* <li>Default to <code>"AES/CBC/PKCS5Padding"</code> if this key is absent.</li>
* </ul>
* </tr>
* <tr>
* <td><b>x-amz-tag-len</b></td>
* <td>Tag length (in bits) when AEAD is in use.
* <ul>
* <li>Only applicable if AEAD is in use. This meta information is absent otherwise, or if KMS is in use.</li>
* <li>Supported value: <code>"128"</code></li>
* </ul>
* </tr>
* </table>

<p>
<h3>KMS Integration for client-side encryption</h3>
* <ol>
* <li>All client-side KMS protected S3 objects are stored in v2 crypto meta information format. However, the key wrapping algorithm, <b>x-amz-wrap-alg</b> is always set to <b>"kms"</b>.</li>
* <li>The KMS Customer Master Key ID is currently stored as part of the material description, <b>x-amz-matdesc</b>, under the key-name
* `<b>`"kms_cmk_id"`/`</b>&lt;/li&gt;  
* `</ol>`
* `</p>`
* `<h4>`Sample S3 crypto meta information for KMS client-side encryption`/`</h4>`
* `<p>`
* `<pre>`
* `x-amz-meta-x-amz-key-v2:`
* CiAkJ2GqwpJRnt4izuwOaIC2YdNRNSMElG5rw3pEI+J8hKnaAQEBAwB4CdhqsKSUZ7eI+7o8KGpQtmHTUTUjBLJRu8N6RPCiflAACAB+MHwGCSqGSIb3DQEHBqBvMG0CAQAwaAYJKoZIhvcNAMQECBMB4GCWCGSAFlAwQBLjARRAyDOrzN3KHViWixDVcCARCAOwhZ6xA8rob3Z0mN3uKFieHKVyiC/PKlckALvGl0VLnf86m cZJNpTXtwMkkKjDocVj3ZlzGjZTuFs0OH
* `x-amz-iv: xqOHBh+M6y0UogwG5SHJM3Q==`  
* `x-amz-unencrypted-content-length: 32768`  
* `x-amz-cek-alg: AES/CBC/PKCS5Padding`  
* `x-amz-wrap-alg: kms`  
* `x-amz-matdesc: {"kms_cmk_id":"d3fd2273-4ca0-4da5-b1c2-c89742ecc6a26"}`
* `</pre>`
* `<h5>`CryptoMode: Authenticated Encryption`/`</h5>`
* `<pre>`
* `x-amz-key-v2:`
* CiAkJ2GqwpJRnt4izuwOaIC2YdNRNSMElG5rw3pEI+J8hKnaAQEBAwB4CdhqsKSUZ7eI+7o8KGpQtmHTUTUjBLJRu8N6RPCiflAACAB+MHwGCSqGSIb3DQEHBqBvMG0CAQAwaAYJKoZIhvcNAMQECBMB4GCWCGSAFlAwQBLjARRAyFVvyWj6QmwePK1E5CARCAO1VzoM6KxaA9DHyMMin/BEbbjle2vLA7IY7f9OkUoSqrjxKxpnID9QaL/7928m1Djld0OShfHejfm89C
* `x-amz-iv: VZBksyij6DhrUMye`  
* `x-amz-unencrypted-content-length: 32768`  
* `x-amz-cek-alg: AES/GCM/NoPadding`  
* `x-amz-wrap-alg: kms`  
* `x-amz-matdesc: {"kms_cmk_id":"d3fd2273-4ca0-4da5-b1c2-c89742ecc6a26"}`
* `x-amz-tag-len: 128`  
* `</pre>`
* `<h5>`S3 metadata of an instruction file`/`</h5>`
* `<p>`
* The following key is always included as part of the S3 metadata of the instruction file (which is just an S3 object). Since the SDK treats this key as "user metadata", the key name is always prefixed with `<b>`x-amz-meta-`/`</b>` when physically stored.
* `<p>`
* `<table border=1>`
* `<tr aligned="left">`
* `<th>`key`</th>`
* `<th>`description`</th>`
* </tr>
* </tr>
* <td width="200"><b>x-amz-crypto-instr-file</b></td>
* </tr>
* the presence of this key is used to indicate this S3 object is an
* instruction file (of an associated client-side encrypted S3 object).</td>
* </tr>
* </table>
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/package-info.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
* for applicable license terms and NOTICE.txt for applicable notices.
* Licensed under the Apache License, Version 2.0 (the "License”).
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/crypto/SecuredCEK.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/crypto/KMSSecuredCEK.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License”).
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/auth/DecodedStreamBuffer.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/S3ObjectIds.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/S3Event.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/SSEAwsKeyManagementParams.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/PutInstructionFileRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/SimpleMaterialProvider.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/InstructionFileId.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/NotificationConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/SSECustomerKeyProvider.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/internal/FileDeletionEvent.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/NotificationConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/SSEAwsKeyManagementParams.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/NotificationConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/S3DataSource.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/PutObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/internal/AmazonS3ExceptionBuilder.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/internal/MultiFileOutputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/KMSEncryptionMaterialsProvider.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/UploadObjectObserver.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/TopicConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/OnFileDelete.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/internal/S3Direct.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
  jar/com/amazonaws/services/s3/model/EncryptedPutObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/internal/S3SyncProgressListener.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/MultipleFileTransferStateChangeListener.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3DirectSpi.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/KeyWrapException.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/MaterialsDescriptionProvider.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/PartCreationEvent.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/EncryptedGetObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/UploadObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/AbstractPutObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/EncryptionMaterialsFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3URI.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/EncryptedInitiateMultipartUploadRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ExtraMaterialsDescription.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/CloudFunctionConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SSECustomerKey.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/S3ObjectIIdBuilder.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/QueueConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/auth/ChunkContentIterator.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SSEAwsKeyManagementParamsProvider.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*/
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/PauseResult.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights
* Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/auth/S3SignerProvider.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
* http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
 * Gets the optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin returning
 * results. Results are always ordered first lexicographically (i.e.
 * alphabetically) and then from most recent version to least recent
 * version.
 * <p>
 * A key marker must be specified when specifying a <code>versionIdMarker</code>.
 * Results begin immediately after the version with the specified key and
 * version ID.
 * <p>
 * @return The optional <code>versionIdMarker</code> parameter indicating where in the
 * sorted list of all versions in the specified bucket to begin
 * returning results.
 *
 * @see ListVersionsRequest#setVersionIdMarker(String)
 * @see ListVersionsRequest#withVersionIdMarker(String)
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetBucketWebsiteConfigurationRequest.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 * 
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetBucketWebsiteConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3ClientParamsWrapper.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/TransferManagerParams.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/RoutingRule.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3ClientParams.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/InitiateMultipartUploadHeaderHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ListObjectsV2Result.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3AbortableInputStream.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3RequesterChargedHeaderHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3VersionResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetObjectTaggingResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetObjectAclRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/GetObjectTaggingResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetObjectTaggingRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/Tag.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/Tag.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/Tag.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/DeleteObjectTaggingHeaderHandler.java
jar/com/amazonaws/services/s3/model/GetObjectTaggingRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/ObjectTagging.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/S3AccelerateUnsupported.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/BucketAccelerateStatus.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/AbortIncompleteMultipartUpload.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016-2020 Amazon Technologies, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
*
*    http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/S3RequesterChargedResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/AbortIncompleteMultipartUpload.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
*
*    http://aws.amazon.com/apache2.0
*
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/MD5DigestCalculatingInputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3VersionSummary.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetBucketLifecycleConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ObjectListing.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/MultiFactorAuthentication.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/BucketVersioningConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/DeleteVersionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetBucketTaggingConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetBucketLoggingConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GroupGrantee.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/package-info.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Permission.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetBucketCrossOriginConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3ObjectSummary.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

// The Amazon S3 DevPay token header is a special exception and can be safely moved
Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/AmazonS3Client.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/StaticEncryptionMaterialsProvider.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/iterable/S3Versions.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/Download.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/DownloadImpl.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/iterable/S3Objects.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
* Licensed under the Apache License, Version 2.0 (the "License").
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/
/**
 * Interface for providing encryption materials.
 * Implementations are free to use any strategy for providing encryption
 * materials, such as simply providing static material that doesn't change,
 * or more complicated implementations, such as integrating with existing
 * key management systems.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/EncryptionMaterialsProvider.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
 * for applicable license terms and NOTICE.txt for applicable notices.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/SetObjectTaggingResponseHeaderHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/GetObjectTaggingResponseHeaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/S3VersionHeaderHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/RequestPaymentConfigurationXmlFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/RestoreObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/auth/policy/resources/S3ObjectResource.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/AnalyticsPredicateVisitorImpl.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/CompleteMultipartUploadRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PartSummary.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/S3MetadataResponseHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/JceEncryptionConstants.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/MetadataDirective.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/MetadataEntry.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Region.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ProgressListener.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/UploadCallable.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetBucketPolicyStatusResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/QuoteFields.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PartETag.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/UploadPartResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/DeleteObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PartETag.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3ArnConverter.java
Open Source Used In DNAC Platform Arica  11762

*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/transfer/package-info.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/SetPublicAccessBlockResult.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/transfer/internal/TransferProgressUpdatingListener.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/GetRequestPaymentConfigurationRequest.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/model/ReplicationTimeStatus.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/ReplicationTimeValue.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/CannedAccessControlList.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/S3ResourceType.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/PartListing.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/GetObjectResult.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/transfer/TransferManagerConfiguration.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/internal/S3RestoreOutputPathHeaderHandler.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/internal/ObjectExpirationHeaderHandler.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/S3Resource.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/CSVInput.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/ListBucketsRequest.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/transfer/Upload.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/ObjectTaggingProvider.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/LifecyclePredicateVisitorImpl.java
*/opt/cola/permissions/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1.11-704-sources.jar/com/amazonaws/services/s3/model/OutputLocation.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-1-11-704-sources-jar/com/amazonaws/services/s3/model/CSVOutput.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-1-11-704-sources-jar/com/amazonaws/services/s3/internal/crypto/AdjustedRangeInputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/BucketConfigurationXmlFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PolicyStatus.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/auth/policy/resources/S3BucketResource.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/MetricsStatus.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ExpressionType.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ReplicationTime.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ReplicationPredicateVisitorImpl.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ExpressionType.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3XmlResponseHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/CopyCallable.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SetRequestPaymentConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ListMultipartUploadsRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/S3ObjectResource.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/TransferProgress.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/CopyObjectRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/HeadBucketRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3ClientOptions.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/CompleteMultipartCopy.java

Open Source Used in DNAC Platform Arica 11764
Open Source Used In DNAC Platform Arica 11765

* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/S3Location.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/internal/UseArnRegionResolver.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/ListPartsRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/ListObjectsRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/ObjectMetadata.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/CopyObjectResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/CryptoConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/model/PublicAccessBlockConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/transfer/TransferManager.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/transfer/internal/TransferManagerUtils.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources.jar/com/amazonaws/services/s3/transfer/ObjectMetadataProvider.java
Open Source Used In DNAC Platform Arica 11766

* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/AclXmlFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/auth/policy/actions/S3Actions.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/Serialization.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/AbstractSSEHandler.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/RequestXmlFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/UploadImpl.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/DeleteBucketRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/ProgressEvent.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/S3RestoreOutputPathResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/auth/AwsChunkedEncodingInputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/LegacyS3ProgressListener.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/Metrics.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/CompleteMultipartUploadResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/CompleteMultipartUpload.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/Unmarshallers.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/CryptoStorageMode.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-3.1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectResult.java
/*
 * Copyright 2012-2020 Amazon Technologies, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at:
 *
 * http://aws.amazon.com/apache2.0
 *
 * This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
 * OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/Internal/S3ObjectResponseHandler.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/BucketConfigurationXmlFactoryFunctions.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/OutputSerialization.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/AbortMultipartUploadRequest.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GeneratePresignedUrlRequest.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/S3AccessPointResource.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/MultipartUpload.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/UploadContext.java

Found in path(s):
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/DownloadMonitor.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/MultipartUpload.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/ MultipleFileDownload.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/ MultipleFileUpload.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/ MultipleFileUploadImpl.java
/ *opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/AbstractTransfer.java
* You may not use this file except in compliance with the License.
* A copy of the License is located at
* http://aws.amazon.com/apache2.0
* or in the "license" file accompanying this file. This file is distributed
* on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing
* permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SelectObjectContentEventException.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/transform/ObjectLockRetentionXmlFactory.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/SelectRecordsInputStream.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ObjectLockRetentionMode.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/Progress.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PresignedUrlDownloadConfig.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectLegalHoldResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/ObjectLockRetention.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PresignedUrlUploadResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/JSONOutput.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/PresignedUrlDownloadResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionResult.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/transfer/internal/AbstractDownloadCallableConfig.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-jar/com/amazonaws/services/s3/model/GetObjectRetentionRequest.java
/*

* Copyright 2014-2020 Amazon Technologies, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at:
* http://aws.amazon.com/apache2.0
*
* This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES
* OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
*/

No license file was found, but licenses were detected in source scan.
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

Portions copyright 2006-2009 James Murty. Please see LICENSE.txt
for applicable license terms and NOTICE.txt for applicable notices.

The Apache Software License, Version 2.0
http://www.apache.org/licenses/LICENSE-2.0

Copyright 2015-2020 Amazon.com, Inc. or its affiliates. All Rights Reserved.
Licensed under the Apache License, Version 2.0 (the "License").
You may not use this file except in compliance with the License.
A copy of the License is located at
http://aws.amazon.com/apache2.0

or in the "license" file accompanying this file. This file is distributed
on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing
permissions and limitations under the License.

Copyright 2006-2009 James Murty. Please see LICENSE.txt
for applicable license terms and NOTICE.txt for applicable notices.

The Apache Software License, Version 2.0
http://www.apache.org/licenses/LICENSE-2.0
jar/com/amazonaws/services/s3/model/transform/ServerSideEncryptionRuleStaxUnmarshaller.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/internal/CompleteMultipartUploadRetryCondition.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/transform/StatsStaxUnmarshaller.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/Filter.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/S3KeyFilter.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/transform/FilterStaxUnmarshaller.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/LambdaConfiguration.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/SetBucketAccelerateConfigurationRequest.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AbstractAmazonS3.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/model/BucketReplicationConfiguration.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2017. Amazon.com, Inc. or its affiliates. All Rights Reserved.
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 * http://aws.amazon.com/apache2.0
 * or in the "license" file accompanying this file. This file is distributed 
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either 
 * express or implied. See the License for the specific language governing 
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/transfer/KeyFilter.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3EncryptionClientBuilder.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParams.java
* /opt/cola/permits/1137154767_1614023256.44/0/aws-java-sdk-s3-1-11-704-sources-
jar/com/amazonaws/services/s3/AmazonS3EncryptionClientParamsWrapper.java
1.2552 gson 2.8.6

1.2552.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Android Open Source Project
 * Copyright (C) 2012 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/LinkedTreeMap.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/LinkedHashTreeMap.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Gson authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-
Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/CollectionTypeAdapterFactory.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/TypeAdapters.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/JsonTreeReader.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/ArrayTypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/ObjectTypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/MapTypeAdapterFactory.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson>TypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/JsonReaderInternalAccess.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/JsonTreeWriter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson-TypeAdapterFactory.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/ReflectiveTypeAdapterFactory.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/internal/bind/AdapterRuntimeTypeWrapper.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/JsonParseException.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/JsonPrimitive.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1.jar/com/google/gson/ExclusionStrategy.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-
  jar/com/google/gson/annotations/JsonAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-
  jar/com/google/gson/internal/bind/JsonAdapterAnnotationTypeAdapterFactory.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2008 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-
  jar/com/google/gson/internal/$Gson$Types.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/stream/JsonToken.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/stream/JsonWriter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/stream/JsonScope.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/stream/MalformedJsonException.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/stream/JsonReader.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/LazilyParsedNumber.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/bind/SqlDateTypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/bind/TimeTypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/bind/DateTypeAdapter.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/UnsafeAllocator.java
jar/com/google/gson/internal/bind/TreeTypeAdapter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/JsonSyntaxException.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/internal/Streams.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/FieldAttributes.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/JsonParser.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/LongSerializationPolicy.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/JsonStreamParser.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/FieldAttributes.java
* /opt/cola/permits/1108597571_1606176527.27/0/gson-2-8-6-sources-1-jar/com/google/gson/JsonParser.java
1.2553 dropwizard-metrics-hadoop-metrics2-reporter 0.1.2

1.2553.1 Available under license:

Copyright 2016 Josh Elser

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2554 parse-duration 0.1.2 0.1.2
1.2554.1 Available under license :

The MIT License

Copyright (c) 2013 Jake Rosoman <jkroso@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2555 buffer-crc32 0.2.13 0.2.13

1.2555.1 Available under license:
The MIT License

Copyright (c) 2013 Brian J. Brennan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2556 platform 1.3.1 1.3.1

1.2556.1 Available under license:
Copyright 2014-2016 Benjamin Tan <https://d10.github.io/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
1.2557 object.assign 4.1.0

1.2557.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2558 alsa 1.1.0-0ubuntu1

1.2558.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.
In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or
link a “work that uses the Library” with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable “work that
uses the Library”, as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.2559 native-promise-only 0.8.1 0.8.1
1.2559.1 Available under license :
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2560 bytes 2.3.0 2.3.0
1.2560.1 Available under license :
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Jed Watson <jed.watson@me.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2561 aws4 1.11.0 1.11.0
1.2561.1 Available under license:
Copyright 2013 Michael Hart (michael.hart.au@gmail.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2562 @babel/highlight 7.0.0
1.2562.1 Available under license:
MIT License

Copyright (c) 2014-2018 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2563 @babel/helper-simple-access 7.16.0

1.2563.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
1.2564 lodash.flattendeep 4.4.0

1.2564.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2565 joni 2.1.31

1.2565.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
* Permission is hereby granted, free of charge, to any person obtaining a copy of
* this software and associated documentation files (the "Software"), to deal in
* the Software without restriction, including without limitation the rights to
* use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
* of the Software, and to permit persons to whom the Software is furnished to do
* so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/constants/internal/StackType.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/CaptureTreeNode.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/BitStatus.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/SyntaxException.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/UnsetAddrList.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/constants/internal/Traverse.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/constants/internal/OPSize.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/UnsetAddrList.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/Traverse.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/OPSize.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/StackType.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/BitStatus.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/SyntaxException.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/UnsetAddrList.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/Traverse.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/OPSize.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/StackType.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/BitStatus.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/SyntaxException.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/UnsetAddrList.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/Traverse.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/OPSize.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/StackType.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/BitStatus.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/SyntaxException.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/UnsetAddrList.java
* /opt/ws_local/PERMITS_SQL/1045445922_1591004900.63/0/joni-2-1-31-sources-jar/org/joni/exception/Traverse.java
Open Source Used In DNAC Platform Arica 11811
**1.2566 proxy-from-env 1.1.0 1.1.0**

**1.2566.1 Available under license:**

The MIT License

Copyright (C) 2016-2018 Rob Wu <rob@robwu.nl>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

**1.2567 xml-apis-ext 1.3.04**

**1.2567.1 Available under license:**

xml-commons/java/external/README.dom.txt $Id: README.dom.txt 477038 2006-11-20 04:40:36Z mrglavas$

HEAR YE, HEAR YE!

All of the .java software and associated documentation about the DOM in this repository are distributed under the license from the W3C, which is provided herein.

LICENSE.dom-software.txt covers all software from the W3C including the following items in the xml-commons project:

xml-commons/java/external/src/org/w3c
and all subdirectories
(Note: SAC (The Simple API for CSS) has been published under an older version of the W3C license. The original license file is LICENSE.sac.html.)

LICENSE.dom-documentation.txt covers all documentation from the W3C including the following items in the xml-commons project:

xml-commons/java/external/xdocs/dom
and all subdirectories

The actual DOM Java Language Binding classes in xml-commons came from:
http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.html

The specification of DOM Level 3's various parts is at:
http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/
http://www.w3.org/TR/2004/REC-DOM-Level-3-LS-20040407/
http://www.w3.org/TR/2004/NOTE-DOM-Level-3-XPath-20040226/

The specification of DOM Level 2's various parts is at:
http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Traversal-Range-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Views-20001113/

The specification of DOM Level 1's various parts is at:

Links to all available W3C DOM Java Bindings can be found at:
http://www.w3.org/DOM/DOMTR

The actual classes of The Simple API for CSS (SAC) came from:
http://www.w3.org/Style/CSS/SAC/
http://www.w3.org/2002/06/sacjava-1.3.zip

The actual DOM Java Language Binding classes for SMIL came from:
http://dev.w3.org/cvsweb/java/classes/org/w3c/dom/smil/
(both ElementTimeControl.java and TimeEvent.java were taken at revision 1.1)

The actual DOM Java Language Binding classes for SVG 1.1 came from:
http://www.w3.org/TR/SVG11/java.html

========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache xml-commons xml-apis ==
== distribution.                                                      ==
========================================================================

Apache XML Commons XML APIs
Copyright 2006 The Apache Software Foundation.
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)
THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>


This license came from: http://www.w3.org/Consortium/Legal/copyright-documents-20021231

W3C DOCUMENT LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231

Public documents on the W3C site are provided by the copyright holders under the following license. By using and/or copying this document, or the W3C document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C
document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [Sdate-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

-------------------------------------------------------------------

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of
the license, and removes the ambiguous grant of "use". See the older
formulation for the policy prior to this date. Please see our Copyright FAQ for
common questions about using materials from our site, such as the translating
or annotating specifications. Other questions about this notice can be directed
to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) $

1.2568 esprima 4.0.1 4.0.1

1.2568.1 Available under license :

Copyright JS Foundation and other contributors, https://js.foundation/

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2569 parquet-encoding 1.10.0

1.2569.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This project includes code from Daniel Lemire's JavaFastPFOR project in this binary artifact. The "Lemire" bit packing classes produced by parquet-generator are derived from the JavaFastPFOR project.

Copyright: 2013 Daniel Lemire
Home page: http://lemire.me/en/
Project page: https://github.com/lemire/JavaFastPFOR
License: Apache License Version 2.0 http://www.apache.org/licenses/LICENSE-2.0

1.2570 js-base64 2.6.4 2.6.4
1.2570.1 Available under license :

Copyright (c) 2014, Dan Kogai
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of {{project}} nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2571 supports-color 2.0.0

1.2571.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2572 guava 30.1-jre
1.2572.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Ascii.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Strings.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Equivalence.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/Beta.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java
/*
* Copyright (C) 2016 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ElementOrder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/Comparators.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/DirectedMultiNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/AbstractGraphBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/NetworkBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/StandardMutableGraph.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2020 The Guava Authors
 */

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
 */
/**
 * Holder for web specializations of methods of {@code Doubles}. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/DoublesMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2020 The Guava Authors
 */

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)

* Unless required by applicable law or agreed to in writing, software distributed under the License
  * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
  * or implied. See the License for the specific language governing permissions and limitations under
  * the License.

*/
/**
* Holder for web specializations of methods of `Ints`. Intended to be empty for regular
* version.
*/

Found in path(s):
* `/opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/IntsMethodsForWeb.java`
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* `/opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collect/ImmutableSortedAsList.java`
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2015 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* [http://www.apache.org/licenses/LICENSE-2.0](http://www.apache.org/licenses/LICENSE-2.0)
* Unless required by applicable law or agreed to in writing, software
Open Source Used In DNAC Platform Arica 11834

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/CollectSpliterators.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ImmutableBiMapFauxverideShim.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/AbstractMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ForwardingMapEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/AbstractIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/Sets.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ForwardingMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/Multimaps.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/BitMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/Synchronized.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ComparatorOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/CompoundOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/AbstractMapBasedMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingSortedMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/MutableClassToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/AbstractMapBasedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/UsingToStringOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/Iterables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/TreeMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/NaturalOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/Lists.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/RegularImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/LinkedHashSetMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ArrayListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ByFunctionOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ReverseOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractMapEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/LexicographicalOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/LinkedHashSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/HashMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingCollection.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ClassToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Maps.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingObject.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/MapDifference.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/EnumBiMap.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/Resources.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Predicates.java
jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/Closeables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/EventBus.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Suppliers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/DirectExecutor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/AllowConcurrentEvents.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Defaults.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/CountingOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ListenableFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Functions.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteStreams.java

No license file was found, but licenses were detected in source scan.

/*
   * Copyright (C) 2017 The Guava Authors
   *
   * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
   * in compliance with the License. You may obtain a copy of the License at
*/
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ForwardingLock.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/AbstractHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ForwardingCondition.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/Java8Compatibility.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/OverflowAvoidingLockSupport.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/math/ToDoubleRounder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/math/BigDecimalMath.java
  * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/Java8Compatibility.java
No license file was found, but licenses were detected in source scan.

/ *
  * Copyright (C) 2015 The Guava Authors
  *
  * Licensed under the Apache License, Version 2.0 (the "License"); you
  * may not use this file except in compliance with the License. You may
  * obtain a copy of the License at
  *
  * http://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software
  * distributed under the License is distributed on an "AS IS" BASIS,
  * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
  * implied. See the License for the specific language governing
  * permissions and limitations under the License.
  */

Found in path(s):
  * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/collect/Streams.java
No license file was found, but licenses were detected in source scan.

/*
  * Copyright (C) 2019 The Guava Authors
  *
  * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
  * in compliance with the License. You may obtain a copy of the License at
  *
  * http://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software distributed under the License
  * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
  * or implied. See the License for the specific language governing permissions and limitations under
  * the License.
  */

Found in path(s):
  * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/Platform.java
  * /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/Internal.java
No license file was found, but licenses were detected in source scan.
OPEN SOURCE USED IN DNAC PLATFORM

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/LittleEndianByteArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/ReaderInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collect/ConsumingQueueIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/InterruptibleTask.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AggregateFutureState.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either

express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/JdkBackedImmutableMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/JdkBackedImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/ImmutableSupplier.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ExecutionSequencer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/IgnoreJRERequirement.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under

* the License.
*/

/*
 * This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
 * and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 *
 * As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
 * java.util.HashMap class.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/Striped.java
No license file was found, but licenses were detected in source scan.

/**
 *
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/AbstractCompositeHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/Hashing.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/Optional.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/Ticker.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1119092615_1608716259_54/0/guava-30-1-jre-sources-1.jar/com/google/common/base/PairwiseEquivalence.java
Open Source Used In DNAC Platform Arica

11846
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedMultisetFauxverideShim.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/GeneralRange.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedIterable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RangeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractRangeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Count.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedIterables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedMultiset.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2007 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
>Returns an array containing all of the elements in the specified collection. This method
returns the elements in the order they are returned by the collection's iterator. The returned
array is "safe" in that no references to it are maintained by the collection. The caller is
thus free to modify the returned array.

This method assumes that the collection size doesn't change while the method is running.

TODO(kevinb): support concurrently modified collections?

@param c the collection for which to return an array of elements

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/collection/ObjectArrays.java
No license file was found, but licenses were detected in source scan.

* Copyright (C) 2012 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License
is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
or implied. See the License for the specific language governing permissions and limitations under
the License.

This method was rewritten in Java from an intermediate step of the Murmur hash function in
http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
following header:

MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
hereby disclaims copyright to this source code.

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/SubscriberRegistry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/Dispatcher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/base/MoreObjects.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ListenerCallQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/math/Quantiles.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/eventbus/Subscriber.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2006 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 */
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ServiceManagerBridge.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Java8Usage.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractTable.java

Copyright (C) 2013 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License
is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
or implied. See the License for the specific language governing permissions and limitations under
the License.

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractTable.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AllEqualOrdering.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TransformedListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/FilteredEntrySetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SortedMultisetBridge.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactLinkedHashSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/CompactLinkedHashMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableEnumMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingImmutableList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingBlockingDeque.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingDeque.java

Open Source Used In DNAC Platform Arica  11854
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/TransformedIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingImmutableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/TreeTraverser.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/CompactHashSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/DescendingMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/FilteredEntryMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/RangeMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingNavigableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/TreeRangeMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/DescendingImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/AbstractMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/RegularImmutableAsList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/EvictingQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingNavigableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/ForwardingImmutableSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collection/AbstractNavigableMap.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2016 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of { @code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2007 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
* This following method is a modified version of one found in
* http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
* which contained the following notice:
*
Open Source Used In DNAC Platform Arica 11857

* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
  the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
*/

Found in path(s):
  */opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2013 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
  */opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/ImmutableMapEntry.java
  */opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/MultimapBuilder.java
  */opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/MoreFiles.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/Tables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/Serialization.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/StandardTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/PeekingIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/RegularImmutableMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/SingletonImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/RegularImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableBiMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableMapValues.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/Collections2.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableMapEntrySet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/ImmutableCollection.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/Table.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/EmptyImmutableListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/CollectionPreconditions.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collections/HashBasedTable.java
/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for web specializations of methods of [ @code Shorts ]. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/primitives/ShortsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/Closer.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/Stats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/html/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/PairStatsAccumulator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/Element.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1.jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/SipHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/math/PairedStats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/xml/package-info.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/ServiceManager.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2019 The Guava Authors
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/IncidentEdgeSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/CompactHashing.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/graph/Traverser.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ClosingFuture.java

No license file was found, but licenses were detected in source scan.

/
/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Outer class that exists solely to let us write [code Partially.GwtIncompatible] instead of plain
* [code GwtIncompatible]. This is more accurate for [link Futures#catching], which is available
* under GWT but with a slightly different signature.
*
* <p>We can't use [code PartiallyGwtIncompatible] because then the GWT compiler wouldn't recognize
* it as a [code GwtIncompatible] annotation. And for [code Futures.catching], we need the GWT
* compiler to autostrip the normal server method in order to expose the special, inherited GWT
* version.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/util/concurrent/Partially.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/collection/TopKSelector.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/Graphs.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/io/RecursiveDeleteOption.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/graph/Network.java
No license file was found, but licenses were detected in source scan.

/
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/AtomicDoubleArray.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/LongAdder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/hash/Striped64.java
No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2008 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/FileBackedOutputStream.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Shorts.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Doubles.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/SequentialExecutor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/CharMatcher.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Booleans.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Chars.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Floats.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Bytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/MultiReader.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Stopwatch.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Converter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Longs.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/Ints.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/internal/Finalizer.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/InetAddresses.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/Escaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/thirdparty/publicsuffix/TrieParser.java

* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/MinMaxPriorityQueue.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/AbstractSequentialIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingListMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RowSortedTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/ForwardingImmutableCollection.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/UnmodifiableListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/SortedMapDifference.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/SortedMultiset.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/SortedMultisets.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/collection/DenseImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
  jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-

jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/GwtIncompatible.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SparseImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/MapMakerInternalMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/MapMaker.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/Cut.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingFluentFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Platform.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-
jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collect/RegularImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/util/concurrent/AbstractService.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/base/Splitter.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/escape/Escapers.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
* Not supported. <b>You are attempting to create a map that may contain a non-{@code Comparable} key.</b>
* Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy version.
* @throws UnsupportedOperationException always
* @deprecated Pass a key of type `Comparable` to use `ImmutableSortedMap#of(Comparable, Object)`.

*/

Found in path(s):
*/

*/

* Copyright (C) 2020 The Guava Authors
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*/

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

*/

*/

* Copyright (C) 2009 The Guava Authors
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*/

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ComputationException.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/DiscreteDomain.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableAsList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ArrayTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ForwardingTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/TableCollectors.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableEnumSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableSortedSet.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSortedMap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SingletonImmutableList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/SingletonImmutableTable.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ImmutableSetMultimap.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/ComparisonChain.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/RegularImmutableList.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/AbstractIndexedListIterator.java
* /opt/cola/permits/1119092615_1608716259.54/0/guava-30-1-jre-sources-1-jar/com/google/common/collection/EmptyImmutableSetMultimap.java

1.2573 audience-annotations 0.5.0
1.2573.1 Available under license :

Apache Yetus - Audience Annotations
Copyright 2015-2017 The Apache Software Foundation
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[[]]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2574 latest-version 5.1.0

1.2574.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2575 json5 0.5.1 0.5.1

1.2575.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2576 jackson-dataformat-yaml 2.4.2

1.2576.1 Available under license:

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.
This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:
1.2577 statuses 1.5.0 1.5.0

1.2577.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2016 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2578 jetty-servlets 9.4.41.v20210516

1.2578.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates'
from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not
include additions to the Program which: (i) are separate modules of
software distributed in conjunction with the Program under their own
license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free copyright license to
   reproduce, prepare derivative works of, publicly display, publicly
   perform, distribute and sublicense the Contribution of such Contributor,
   if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
   Recipient a non-exclusive, worldwide, royalty-free patent license under
   Licensed Patents to make, use, sell, offer to sell, import and otherwise
   transfer the Contribution of such Contributor, if any, in source code and
   object code form. This patent license shall apply to the combination of
   the Contribution and the Program if, at the time the Contribution is
   added by the Contributor, such addition of the Contribution causes such
   combination to be covered by the Licensed Patents. The patent license
   shall not apply to any other combinations which include the Contribution.
   No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION
LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under
applicable law, it shall not affect the validity or enforceability of the
remainder of the terms of this Agreement, and without further action by the
parties hereto, such provision shall be reformed to the minimum extent
necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware)
infringes such Recipient's patent(s), then such Recipient's rights granted
under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to
comply with any of the material terms or conditions of this Agreement and does
not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement
and any licenses granted by Recipient relating to the Program shall continue
and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be
modified in the following manner. The Agreement Steward reserves the right to
publish new versions (including revisions) of this Agreement from time to
time. No one other than the Agreement Steward has the right to modify this
Agreement. The Eclipse Foundation is the initial Agreement Steward. The
Eclipse Foundation may assign the responsibility to serve as the Agreement
Steward to a suitable separate entity. Each new version of the Agreement will
be given a distinguishing version number. The Program (including
Contributions) may always be distributed subject to the version of the
Agreement under which it was received. In addition, after a new version of the
Agreement is published, Contributor may elect to distribute the Program
(including its Contributions) under the new version. Except as expressly
stated in Sections 2(a) and 2(b) above, Recipient receives no rights or
licenses to the intellectual property of any Contributor under this Agreement,
whether expressly, by implication, estoppel or otherwise. All rights in the
Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
  * org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
  * org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl
The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.2579 safe-buffer 5.1.1 5.1.1

1.2579.1 Available under license:
The MIT License (MIT)

Copyright (c) Feross Aboukhadijeh
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2580 protobuf-java 3.13.0

1.2580.1 Available under license:

No license file was found, but licenses were detected in source scan.

// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/google/protobuf/type.proto
* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/NewInstanceSchemas.java
* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/CodedInputStreamReader.java
* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/TextFormatParseLocation.java
Open Source Used In DNAC Platform Arica 11894
Open Source Used In DNAC Platform Arica 11895

/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/TextFormatEscaper.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/CodedInputStream.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/AbstractProtobufList.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/Descriptors.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/JavaType.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/ExtensionRegistry.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/Writer.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/google/protobuf/api.proto */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/UnsafeUtil.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/ManifestSchemaFactory.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/RepeatedFieldBuilderV3.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/BlockingService.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/CodedOutputStream.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/ExtensionRegistryFactory.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/RopeByteString.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/FieldSet.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/MessageReflection.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/ListFieldSchema.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/LazyStringList.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/SchemaFactory.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/Protobuf.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/Android.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/AbstractMessageLite.java */
/* /opt/ws_local/PERMITS_SQL/1092310281_1600708753.46/0/protobuf-java-3-13-0-sources-jar/com/google/protobuf/GeneratedMessageLite.java */
Open Source Used In DNAC Platform Arica  11896
1.2581 commons-pool 2.6.1

1.2582 jetty-continuation 9.3.27.v20190418

1.2582.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved. This program and the accompanying materials
// are made available under the terms of the Eclipse Public License v1.0
// and Apache License v2.0 which accompanies this distribution.
// The Eclipse Public License is available at
// The Apache License v2.0 is available at
// You may elect to redistribute this code under either of these licenses.

Found in path(s):
* /opt/cola/permits/1136447697_1613767081.95/0/jetty-continuation-9-3-27-v20190418-sources-1-jar/org/eclipse/jetty/continuation/ContinuationSupport.java
* /opt/cola/permits/1136447697_1613767081.95/0/jetty-continuation-9-3-27-v20190418-sources-1-jar/org/eclipse/jetty/continuation/Servlet3Continuation.java
* /opt/cola/permits/1136447697_1613767081.95/0/jetty-continuation-9-3-27-v20190418-sources-1-jar/org/eclipse/jetty/continuation/FauxContinuation.java
* /opt/cola/permits/1136447697_1613767081.95/0/jetty-continuation-9-3-27-v20190418-sources-1-jar/org/eclipse/jetty/continuation/ContinuationThrowable.java
* /opt/cola/permits/1136447697_1613767081.95/0/jetty-continuation-9-3-27-v20190418-sources-1-jar/org/eclipse/jetty/continuation/Continuation.java
The Eclipse Foundation makes available all content in this plug-in ("Content"). The Content is dual licensed and is provided to you under the terms and conditions of the Eclipse Public License Version 1.0 ("EPL") as well as the Apache Software License Version 2.0. A copy of the EPL is available at <a href="http://www.eclipse.org/legal/epl-v10.html">http://www.eclipse.org/legal/epl-v10.html</a>. A copy of the ASL is available at <a href="http://www.apache.org/licenses/LICENSE-2.0.html">http://www.apache.org/licenses/LICENSE-2.0.html</a>. For purposes of the EPL, "Program" will mean the Content.

Permission to use, copy, modify and distribute UnixCrypt granted provided that the copyright notice appears in all copies.

1.2583 estraverse 4.3.0 4.3.0

1.2583.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2584 h2 1.4.197
1.2585 lodash.debounce 4.0.8 4.0.8

1.2585.1 Available under license :

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

### 1.2586 babel-helper-get-function-arity 6.24.1

#### 6.24.1

**1.2586.1 Available under license:**

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 1.2587 jsch 0.1.55

### 1.2588 @babel/plugin-transform-duplicate-keys 7.16.0

#### 1.2588.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2589 yup 0.29.3 0.29.3

1.2589.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Jason Quense

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2590 babel-node 7.16.0
1.2590.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2591 glob 7.1.3
1.2591.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2592 @babel/helper-get-function-arity

7.10.4

1.2592.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2593 openjpeg 1.6.35

1.2593.1 Available under license :

### MIT License

Copyright (c) 2017 by Jim Pattee <jimp03@email.com>.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2002-2014, Universite catholique de Louvain (UCL), Belgium
Copyright (c) 2002-2014, Professor Benoit Macq
Copyright (c) 2001-2003, David Janssens
Copyright (c) 2002-2003, Yannick Verschueren
Copyright (c) 2003-2007, Francois-Olivier Devaux and Antonin Descampe
Copyright (c) 2005, Herve Drolon, FreeImage Team
Copyright (c) 2007, Digital Signal Processing Laboratory, Universit degli studi di Perugia (UPG), Italy
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Little CMS
Copyright (c) 1998-2011 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.

libpng versions 1.0.7, July 1, 2000 through 1.6.25, September 1, 2016 are Copyright (c) 2000-2002, 2004, 2006-2016 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik
libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88,
and are distributed according to the same disclaimer and license as
libpng-0.88, with the following individuals added to the list of
Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners
but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors"
is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors
and Group 42, Inc. disclaim all warranties, expressed or implied,
including, without limitation, the warranties of merchantability and of
fitness for any purpose. The Contributing Authors and Group 42, Inc.
assume no liability for direct, indirect, incidental, special, exemplary,
or consequential damages, which may result from the use of the PNG
Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this
source code, or portions hereof, for any purpose, without fee, subject
to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not
   be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any
   source or altered source distribution.
The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

END OF COPYRIGHT NOTICE, DISCLAIMER, and LICENSE.

TRADEMARK:

The name "libpng" has not been registered by the Copyright owner as a trademark in any jurisdiction. However, because libpng has been distributed and maintained world-wide, continually since 1995, the Copyright owner claims "common-law trademark protection" in any jurisdiction where common-law trademark is recognized.

OSI CERTIFICATION:

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative. OSI has not addressed the additional disclaimers inserted at version 1.0.7.

EXPORT CONTROL:

The Copyright owner believes that the Export Control Classification Number (ECCN) for libpng is EAR99, which means not subject to export controls or International Traffic in Arms Regulations (ITAR) because it is open source, publicly available software, that does not contain any encryption software. See the EAR, paragraphs 734.3(b)(3) and 734.7(b).

Glenn Randers-Pehrson
glennrp at users.sourceforge.net
September 1, 2016
/*
 * The copyright in this software is being made available under the 2-clauses BSD License, included below. This software may be subject to other third party and contributor rights, including patent rights, and no such rights are granted under this license.
 *
 * Copyright (c) 2002-2014, Universite catholique de Louvain (UCL), Belgium
 * Copyright (c) 2002-2014, Professor Benoit Macq
 * Copyright (c) 2003-2014, Antonin Descampe
 * Copyright (c) 2003-2009, Francois-Olivier Devaux
 * Copyright (c) 2005, Herve Drolon, FreeImage Team
 * Copyright (c) 2002-2003, Yannick Verschueren
 * Copyright (c) 2001-2003, David Janssens
1.2594 serve-static 1.10.3 1.10.3

1.2594.1 Available under license :

(The MIT License)

Copyright (c) 2010 Sencha Inc.
Copyright (c) 2011 LearnBoost
Copyright (c) 2011 TJ Holowaychuk
Copyright (c) 2014-2015 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2595 passport-strategy 1.0.0 1.0.0
1.2595.1 Available under license :
(The MIT License)

Copyright (c) 2011-2013 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2596 jszip 3.5.0 3.5.0
1.2596.1 Available under license :
JSZip is dual licensed. You may use it under the MIT license *or* the GPLv3 license.

The MIT License
===============

Copyright (c) 2009-2016 Stuart Knightley, David Duponchel, Franz Buchinger, Antnio Afonso

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

GPL version 3
==============

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.
To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.
"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major
Component, and (b) serves only to enable use of the work with that
Major Component, or to implement a Standard Interface for which an
implementation is available to the public in source code form. A
"Major Component", in this context, means a major essential component
(kernel, window system, and so on) of the specific operating system
(if any) on which the executable work runs, or a compiler used to
produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all
the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to
control those activities. However, it does not include the work's
System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but
which are not part of the work. For example, Corresponding Source
includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically
linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those
subprograms and other parts of the work.

The Corresponding Source need not include anything that users
can regenerate automatically from other parts of the Corresponding
Source.

The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of
copyright on the Program, and are irrevocable provided the stated
conditions are met. This License explicitly affirms your unlimited
permission to run the unmodified Program. The output from running a
covered work is covered by this License only if the output, given its
content, constitutes a covered work. This License acknowledges your
rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not
convey, without conditions so long as your license otherwise remains
in force. You may convey covered works to others for the sole purpose
of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
not control copyright. Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.
If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:
a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the
covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

1.2597 babel-plugin-transform-es2015-
classes 6.24.1
1.2597.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The "Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International.

SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

 Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2598 asm 3.2

1.2598.1 Available under license :

No license file was found, but licenses were detected in source scan.

/***
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2007 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 * contributors may be used to endorse or promote products derived from
 * this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/commons/JSRInlinerAdapter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/JumpInsnNode.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/analysis/SmallSet.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/util/ASMifierMethodVisitor.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/util/ASMifierAbstractVisitor.java
No license file was found, but licenses were detected in source scan.

/**
* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/SAXCodeAdapter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/SAXClassAdapter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/ASMContentHandler.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/SAXAnnotationAdapter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/Processor.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/SAXFieldAdapter.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/SAXAdapter.java

No license file was found, but licenses were detected in source scan.

* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/package.html
No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2007 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/util/ASMifiable.java
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/util/Traceable.java
No license file was found, but licenses were detected in source scan.

ASM XML Adapter
Copyright (c) 2004, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/xml/asm-xml.dtd
No license file was found, but licenses were detected in source scan.
2005 INRIA, France Telecom

* All rights reserved.

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/commons/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/analysis/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/util/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/attrs/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/tree/package.html
* /opt/cola/permits/1136493993_1613774733.35/0/asm-all-3-2-sources-2-jar/org/objectweb/asm/signature/package.html

1.2599 preserve 0.2.0
1.2599.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2600 @babel/plugin-transform-modules-umd 7.16.0

1.2600.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2601 worker-loader 1.1.1 1.1.1

1.2601.1 Available under license:
Copyright JS Foundation and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2602 babel-plugin-transform-es2015-duplicate-keys 6.24.1

1.2602.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software")>> is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to
The following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2603 grpc-protobuf-lite 1.27.1

1.2603.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.
1.2604 csv-parse 4.8.2 4.8.2

1.2604.1 Available under license:

The MIT License (MIT)

Copyright (c) 2010 Adaltas

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2605 internal-slot 1.0.3

1.2605.1 Available under license:

MIT License

Copyright (c) 2019 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2606 append-field 0.1.0 0.1.0
1.2606.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Linus Unnebeck

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2607 avatica 1.17.0
1.2607.1 Available under license:  
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Calcite -- Avatica
Copyright 2012-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.2608 range-parser 1.0.3 1.0.3
1.2608.1 Available under license :
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2609 babel-plugin-proposal-optional-chaining 7.16.0
1.2609.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2610 source-map 0.5.7
1.2610.1 Available under license :

Copyright (c) 2009-2011, Mozilla Foundation and contributors
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the names of the Mozilla Foundation nor the names of project contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2611 memory-pager 1.5.0 1.5.0

1.2611.1 Available under license :

The MIT License (MIT)

Copyright (c) 2017 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2612 eslint-import-resolver-node 0.3.6

1.2612.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Ben Mosher

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2613 @babel/highlight 7.10.4

1.2613.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
1.2614 eyes 0.1.8 0.1.8
1.2614.1 Available under license:

Copyright (c) 2009 cloudhead

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2615 app-root-path 2.0.1 2.0.1
1.2615.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 Chris Morrell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
1.2616 kotlin 1.3.61
1.2616.1 Available under license :

No license file was found, but licenses were detected in source scan.

[ "version":3,"file":"
license/LICENSE.txt file.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
Kotlin.identityHashCode = getObjectHashCode;

Kotlin.Long = function(low, high) {
  /**
   * @param {number} low  The low (signed) 32 bits of the long.
   * @param {number} high  The high (signed) 32 bits of the long.
   * @constructor
   * @final
   */

  /**
   * Returns a Long representation of the given (32-bit) integer value.
   * The fractional part, if any, is rounded down towards zero.
   * Returns zero if this `Double` value is `NaN`, `Long.MIN_VALUE` if it's less than `Long.MIN_VALUE`,
   * `Long.MAX_VALUE` if it's bigger than `Long.MAX_VALUE`.  The number in
   * question.
   * @param {number} value The number in question.
   * @return {!Kotlin.Long} The corresponding Long value.
   */

  /**
   * Returns a Long representation of the given string, written using the given
   * radix.
   * @param {string} str The textual representation of the Long.
   * @param {number=} opt_radix The radix in which the text is written.
   * @return {null}
   */

  /**
   * @param {number} value The 32-bit integer that comes by concatenating
   * the given high and low bits.  Each is assumed to use 32 bits.
   * @return {null}
   */

  /**
   * @param {number} value The 64-bit integer that comes by concatenating
   * the given high and low bits.  Each is assumed to use 32 bits.
   * @return {null}
   */
var result = Kotlin.Long.ZERO;
for (var i = 0; i < str.length; i++) {
  var size = Math.min(8, str.length - i);
  var value = parseInt(str.substring(i, i + size), radix);
  if (size < 8) {
    var power = Kotlin.Long.fromNumber(Math.pow(radix, size));
    result = result.multiply(power).add(Kotlin.Long.fromNumber(value));
  } else {
    result = result.multiply(radixToPower);
    result = result.add(Kotlin.Long.fromNumber(value));
  }
}
return result;

// NOTE: the compiler should inline these constant values below and then remove
// these variables, so there should be no runtime penalty for these.

/**
 * Number used repeated below in calculations.
 * This must appear before the
 * first call to any from* function below.
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_24_DBL_ = Kotlin.Long.TWO_PWR_16_DBL_ * 2;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_32_DBL_ = Kotlin.Long.TWO_PWR_16_DBL_ * 2;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_48_DBL_ = Kotlin.Long.TWO_PWR_32_DBL_ * 2;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_64_DBL_ = Kotlin.Long.TWO_PWR_32_DBL_ * 2;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_63_DBL_ = Kotlin.Long.TWO_PWR_64_DBL_ / 2;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.ZERO = Kotlin.Long.fromInt(0);

/**
 * @type {number}
 * @private
 */
Kotlin.Long.ONE = Kotlin.Long.fromInt(1);

/**
 * @type {number}
 * @private
 */
Kotlin.Long.NEG_ONE = Kotlin.Long.ZERO.negate();

/**
 * @type {number}
 * @private
 */
Kotlin.Long.MAX_VALUE = Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);

/**
 * @type {number}
 * @private
 */
Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);

/**
 * @param {number=} opt_radix The radix in which the text should be written.
 * @return {string} The textual representation of this value.
 * @override
 */
Kotlin.Long.prototype.toString = function(opt_radix) {
  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  if (this.isZero()) {
    return '0';
  }

  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      // We need to change the Long value before it can be negated, so we remove
      // the bottom-most digit in this base and then recurse to do the rest.
      var radixLong = Kotlin.Long.fromNumber(2);
      var result = this.toString(radix) + this.getLowBitsUnsigned().toString(radix);
      return '-' + result;
    } else {
      return '-' + this.negate().toString(radix);
    }
  } else {
    var result = this.toString(radix) + this.getLowBitsUnsigned().toString(radix);
    return result;
  }
};

/**
 * @return {number} The high 32-bits as a signed value.
 */
Kotlin.Long.prototype.getHighBits = function() {
  return this.high_;
};

/**
 * @return {number} The low 32-bits as a signed value.
 */
Kotlin.Long.prototype.getLowBits = function() {
  return this.low_;
};

/**
 * @return {number} The low
 * 32-bits as an unsigned value.
 */
Kotlin.Long.prototype.getLowBitsUnsigned = function() {
  return this.low_;
};
```javascript
/**
 * @return {number} Returns the number of bits needed to represent the absolute value of this Long.
 */
Kotlin.Long.prototype.getNumBitsAbs = function() {
  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) { return 64; }
    else return this.negate().getNumBitsAbs();
  } else {
    var val = this.high_ == 0 ? this.high_ : this.low_;
    for (var bit = 31; bit > 0; bit--) {
      if (((val & (1 << bit)) !== 0) || (this.low_ >> bit & 1 !== 0)) {
        return this.high_ !== 0 ? bit + 33 : bit + 1;
      }
    }
    return 64;
  }
};

/** @return {boolean} Whether this value is zero. */
Kotlin.Long.prototype.isZero = function() {
  return this.high_ == 0 && this.low_ == 0;
};

/** @return {boolean} Whether this value is negative. */
Kotlin.Long.prototype.isNegative = function() {
  return this.high_ < 0;
};

/** @return {boolean} Whether this value is odd. */
Kotlin.Long.prototype.isOdd = function() {
  return (this.low_ & 1) !== 0;
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long equals the other.
 */
Kotlin.Long.prototype.equalsLong = function(other) {
  return (this.high_ == other.high_) && (this.low_ == other.low_);
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long does not equal the other.
 */
Kotlin.Long.prototype.notEqualsLong = function(other) {
  return (this.high_ !== other.high_) || (this.low_ !== other.low_);
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long is less than the other.
 */
Kotlin.Long.prototype.lessThan = function(other) {
  return this.compare(other) < 0;
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long is less than or equal to the other.
 */
Kotlin.Long.prototype.lessThanOrEqual = function(other) {
  return this.compare(other) <= 0;
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long is greater than the other.
 */
Kotlin.Long.prototype.greaterThan = function(other) {
  return this.compare(other) > 0;
};

/**
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long is greater than or equal to the other.
 */
Kotlin.Long.prototype.greaterThanOrEqual = function(other) {
  return this.compare(other) >= 0;
};

/**
 * Compares this Long with the given one.
 * @return {boolean} Whether this Long is greater than the other.
 */
Kotlin.Long.prototype.compare = function(other) {
  if (this.equalsLong(other)) {
    return 0;
  }

  var thisNeg = this.isNegative();
  var otherNeg = other.isNegative();
  if (thisNeg && !otherNeg) {
    return -1;
  }
  if (!thisNeg && otherNeg) {
    return 1;
  }

  // at this point, the signs are the same, so subtraction will not overflow
  if (this.subtract(other).isNegative()) {
    return -1;
  } else {
    return 1;
  }
};

/**
 * Returns the negation of this value.
 * @return {!Kotlin.Long} The negation of this value.
 */
Kotlin.Long.prototype.negate = function() {
  if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
    return other.isOdd() ? Kotlin.Long.MIN_VALUE : Kotlin.Long.ZERO;
  } else if (this.isZero()) {
    return Kotlin.Long.ZERO;
  } else {
    return this.add(Kotlin.Long.ONE);
  }
};

/**
 * Returns the sum of this and the given Long.
 * @param {Kotlin.Long} other Long to add to this one.
 * @return {!Kotlin.Long} The sum of this and the given Long.
 */
Kotlin.Long.prototype.add = function(other) {
  // Divide each number into 4 chunks of 16 bits, and then sum the chunks.
  var a48 = this.high_ >>> 16;
  var a32 = this.high_ & 0xFFFF;
  var a16 = this.low_ >>> 16;
  var a00 = this.low_ & 0xFFFF;

  var b48 = other.high_ >>> 16;
  var b32 = other.high_ & 0xFFFF;
  var b16 = other.low_ >>> 16;
  var b00 = other.low_ & 0xFFFF;

  var c48 = a48 + b48;
  var c32 = a32 + b32;
  var c16 = a16 + b16;
  var c00 = a00 + b00;

  var d48 = (c48 >>> 16) + (a48 & 0xFFFF);
  var d32 = (c32 >>> 16) + (a32 & 0xFFFF);
  var d16 = (c16 >>> 16) + (a16 & 0xFFFF);
  var d00 = (c00 >>> 16) + (a00 & 0xFFFF);

  return Kotlin.Long.fromBits(d48, d32, d16, d00);
};

/**
 * Returns the difference of this and the given Long.
 * @return {!Kotlin.Long} The difference of this and the given Long.
 */
Kotlin.Long.prototype.subtract = function(other) {
  return this.add(other.negate());
};

/**
 * Returns the product of this and the given Long.
 * @param {Kotlin.Long} other Long to multiply with this.
 * @return {!Kotlin.Long} The product of this and the other.
 */
Kotlin.Long.prototype.multiply = function(other) {
  if (this.isZero()) {
    return Kotlin.Long.ZERO;
  } else if (other.isZero()) {
    return Kotlin.Long.ZERO;
  } else {
    return this.add(this.multiply(other));
  }
};
```

(other.equalsLong(Kotlin.Long.MIN_VALUE)) { return this.isOdd() ? Kotlin.Long.MIN_VALUE : Kotlin.Long.ZERO; } else { return this.negate().multiply(other.negate()); }

if (this.isNegative()) {
  if (other.isNegative()) {
    return this.negate().multiply(other.negate());
  } else if (other.isPositive()) {
    return this.multiply(other.negate()).negate();
  }
}

// If both longs are small, use float multiplication
if (this.lessThan(Kotlin.Long.TWO_PWR_24_) && other.lessThan(Kotlin.Long.TWO_PWR_24_)) {
  return Kotlin.Long.fromNumber(this.toNumber() * other.toNumber());
}

// Divide each long into 4 chunks of 16 bits, and then add up 4x4 products.
// We can skip products that would overflow:
var a48 = this.high_ >>> 16;
var a32 = a16 = this.low_ >>> 16;
var a00 = this.low_ & 0xFFFF;
var b48 = other.high_ >>> 16;
var b32 = other.low_ >>> 16;
var b00 = other.low_ & 0xFFFF;

var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
c00 += a00 * b00;
c16 += c00 >>> 16;
c00 &= 0xFFFF;
c16 += a16 * b00;
c32 += c16 >>> 16;
c16 &= 0xFFFF;
c16 += a00 * b16;
c32 += c16 >>> 16;
c16 &= 0xFFFF;
c32 += a16 * b16;
c48 += c32 >>> 16;
c32 &= 0xFFFF;
c32 += a00 * b32;
c48 += c32 >>> 16;
c32 &= 0xFFFF;
c48 += a48 * b00 + a32 * b16 + a16 * b32 + a00 * b48;
c48 &= 0xFFFF;
return Kotlin.Long.fromBits((c16 << 16) | c00, (c48 << 16) | c32);
}

/**
 * Returns this Long modulo the given one.
 * @param {Kotlin.Long} other Long by which to divide.
 * @return {!Kotlin.Long} This Long modulo the given one.
 */
Kotlin.Long.prototype.modulo = function(other) {
  return this.subtract(this.div(other).multiply(other));
};

/** @return {!Kotlin.Long} The bitwise-NOT of this value. */
Kotlin.Long.prototype.not = function() {
  return Kotlin.Long.fromBits(~this.low_, this.high_);
}

// We will tweak the approximate result by changing it in the 48-th digit or
// the smallest non-fractional digit, whichever is larger.
var log2 = Math.ceil(Math.log(approx.toNumber()) / Math.LN2);
var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);
var approxRem = approx.multiply(other);
while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
  approx -= delta;
  approxRem = approxRes.multiply(other);
}

// We know the answer can’t be zero... and actually, zero would cause
// infinite recursion since we would make no progress.
if (approxRes.isZero()) {
  approxRes = Kotlin.Long.ZERO;
}

// Approximate the result of division. This may be a little greater or
// smaller than the actual value.
var approx = Math.max(1, Math.floor(this.toNumber() / other.toNumber()));

// Decrease the approximation until it is smaller than the remainder. Note
// that if it is too large, the product overflows and is negative.
var approxRes = this.multiply(rem.toNumber());
while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
  approx -= delta;
  approxRes = approxRes.multiply(other);
}

// If both longs are small, use float multiplication
if (this.lessThan(Kotlin.Long.TWO_PWR_24_) && other.lessThan(Kotlin.Long.TWO_PWR_24_)) {
  return Kotlin.Long.fromNumber(this.toNumber() * other.toNumber());
}

// Divide each long into 4 chunks of 16 bits, and then add up 4x4 products.
// We can skip products that would overflow:
var a48 = this.high_ >>> 16;
var a32 = a16 = this.low_ >>> 16;
var a00 = this.low_ & 0xFFFF;
var b48 = other.high_ >>> 16;
var b32 = other.low_ >>> 16;
var b00 = other.low_ & 0xFFFF;

var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
c00 += a00 * b00;
c16 += c00 >>> 16;
c00 &= 0xFFFF;
c16 += a16 * b00;
c32 += c16 >>> 16;
c16 &= 0xFFFF;
c16 += a00 * b16;
c32 += c16 >>> 16;
c16 &= 0xFFFF;
c32 += a16 * b16;
c48 += c32 >>> 16;
c32 &= 0xFFFF;
c32 += a00 * b32;
c48 += c32 >>> 16;
c32 &= 0xFFFF;
c48 += a48 * b00 + a32 * b16 + a16 * b32 + a00 * b48;
c48 &= 0xFFFF;
return Kotlin.Long.fromBits(c16 << 16) | c00, (c48 << 16) | c32);
}
-this.high_);\n  this\n
/**
 * Returns the bitwise-AND of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to AND.\n * @return {!Kotlin.Long} The bitwise-AND of this and the other.\n */
Kotlin.Long.prototype.and = function(other) {
  return Kotlin.Long.fromBits(this.low_ & other.low_,
                                 this.high_ & other.high_);
};

/**
 * Returns the bitwise-OR of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to OR.\n * @return {!Kotlin.Long} The bitwise-OR of this and the other.\n */
Kotlin.Long.prototype.or = function(other) {
  return Kotlin.Long.fromBits(this.low_ | other.low_,
                                 this.high_ | other.high_);
};

/**
 * Returns the bitwise-XOR of this Long and the given one.\n * @param {Kotlin.Long} other The Long with which to XOR.\n * @return {!Kotlin.Long} The bitwise-XOR of this and the other.\n */
Kotlin.Long.prototype.xor = function(other) {
  return Kotlin.Long.fromBits(this.low_ ^ other.low_,
                                 this.high_ ^ other.high_);
};

/**
 * Returns this Long with bits shifted to the left by the given amount.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the left by the given amount.\n */
Kotlin.Long.prototype.shiftLeft = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var low = this.low_;
    if (numBits < 32) {
      var high = this.high_;
      return Kotlin.Long.fromBits(
          low << numBits,
          (high << numBits) | (low >>> (32 - numBits)));
    } else {
      return Kotlin.Long.fromBits(0, low << (numBits - 32));
    }
  }
};

/**
 * Returns this Long with bits shifted to the right by the given amount.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the right by the given amount.\n */
Kotlin.Long.prototype.shiftRight = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(
          (low >>> numBits) | (high << (32 - numBits)),
          high >> numBits);
    } else if (numBits == 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >> (numBits - 32), 0);
    }
  }
};

/**
 * Returns this Long with bits shifted to the right by the given amount, with 0s placed into the new leading bits.\n * @param {number} numBits The number of bits by which to shift.\n * @return {!Kotlin.Long} This shifted to the right by the given amount, with 0s placed into the new leading bits.\n */
Kotlin.Long.prototype.shiftRightUnsigned = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(
          (low >>> numBits) | (high << (32 - numBits)),
          high >>> numBits);
    } else if (numBits == 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >>> (numBits - 32), 0);
    }
  }
};

// Support for Kotlin
Kotlin.Long.prototype.equals = function (other) {
    return other instanceof Kotlin.Long &&
    this.equalsLong(other);
};

Kotlin.defineModule = function (id, declaration) {
};

Kotlin.defineInlineFunction = function(tag, fun) {
    return fun;
};

Kotlin.wrapFunction = function(fun) {
    var f = function() {
        f = fun();
        return f.apply(this, arguments);
    };
    return function() {
        return f.apply(this, arguments);
    };
};

Kotlin.isTypeOf = function(type) {
    return function (object) {
        return typeof object === type;
    }
};

Kotlin.isInstanceOf = function (klass) {
    return function (object) {
        return Kotlin.isType(object, klass);
    }
};

Kotlin.orNull = function (fn) {
    return function (object) {
        return object == null ||
        fn(object);
    }
};

Kotlin.andPredicate = function (a, b) {
    return function (object) {
        return a(object) &&
        b(object);
    }
};

Kotlin.kotlinModuleMetadata = function (abiVersion, moduleName, data)
Open Source Used In DNAC Platform Arica  11966

```kotlin
/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> Array<out T>.component5(): T {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun ByteArray.component5(): Byte {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun ShortArray.component5(): Short {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun IntArray.component5(): Int {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun LongArray.component5(): Long {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun FloatArray.component5(): Float {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun DoubleArray.component5(): Double {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun BooleanArray.component5(): Boolean {
    return get(4)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun CharArray.component5(): Char {
    return get(4)
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ByteArray.contains(element: Byte): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun FloatArray.contains(element: Float): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun DoubleArray.contains(element: Double): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun BooleanArray.contains(element: Boolean): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun CharArray.contains(element: Char): Boolean {
    return indexOf(element) >= 0
}
```

```kotlin
/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun <T> Array<out T>.elementAt(index: Int): T

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun ByteArray.elementAt(index: Int): Byte

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun ShortArray.elementAt(index: Int): Short

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun IntArray.elementAt(index: Int): Int

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun LongArray.elementAt(index: Int): Long

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun FloatArray.elementAt(index: Int): Float

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
@sample samples.collections.Collections.Elements.elementAt* 
public expect fun DoubleArray.elementAt(index: Int): Double
```
public expect fun BooleanArray.elementAt(index: Int): Boolean

public expect fun CharArray.elementAt(index: Int): Char

@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun ByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun ShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.elementAtOrNull(index: Int): T? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}

@kotlin.internal.InlineOnly
public inline fun CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNull(index)
}
ShortArray.elementAtOrNull(index: Int): Short? {\n   return this.getOrNull(index)\n}\n/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
IntArray.elementAtOrNull(index: Int): Int? {
   return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
LongArray.elementAtOrNull(index: Int): Long? {\n   return this.getOrNull(index)\n}\n/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
FloatArray.elementAtOrNull(index: Int): Float? {\n   return this.getOrNull(index)\n}\n/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
CharArray.elementAtOrNull(index: Int): Char? {\n   return this.getOrNull(index)\n}\n/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.find(predicate: (T) -> Boolean): T? {\n   return firstOrNull(predicate)\n}\n/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
   return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ShortArray.find(predicate: (Short) -> Boolean): Short? {
   return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
IntArray.find(predicate: (Int) -> Boolean): Int? {
   return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
LongArray.find(predicate: (Long) -> Boolean): Long? {
   return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
FloatArray.find(predicate: (Float) -> Boolean): Float? {
   return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
CharArray.find(predicate: (Char) -> Boolean): Char? {
   return firstOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.findLast(predicate: (T) -> Boolean): T? {\n   return lastOrNull(predicate)\n}\n/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
   return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
   return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.find(predicate: (Double) -> Boolean): Double? {\n   return lastOrNull(predicate)\n}\n/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {
   return firstOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
CharArray.findLast(predicate: (Char) -> Boolean): Char? {
   return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
   return lastOrNull(predicate)
}\n@sample
fun IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such element was found.
 */
fun LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such element was found.
 */
fun FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such element was found.
 */
fun DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such element was found.
 */
fun BooleanArray.findLast(predicate: (Boolean) -> Boolean): Boolean? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such element was found.
 */
fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun IntArray.first(predicate: (Int) -> Boolean): Int? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun LongArray.first(predicate: (Long) -> Boolean): Long? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun FloatArray.first(predicate: (Float) -> Boolean): Float? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun CharArray.first(predicate: (Char) -> Boolean): Char? {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
> Boolean): Long {\n    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun CharArray.first(predicate: (Char) -> Boolean): Char {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun <T> Array<out T>.firstOrNull(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun BooleanArray.firstOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if element was not found.
 */
public inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char? {
    for (element in this) if (predicate(element)) return element
    return null
}
(predicate(element)) return element

return null

*/

* Returns the first element matching the given [predicate], or `null` if element was not found.

*public inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char?

{ for (element in this) if (predicate(element)) return element

return null

*/

* Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*public inline fun <T> Array<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*public inline fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {

return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun ShortArray.getOrNull(index: Int): Short? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun IntArray.getOrNull(index: Int): Int? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun LongArray.getOrNull(index: Int): Long? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun FloatArray.getOrNull(index: Int): Float? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun DoubleArray.getOrNull(index: Int): Double? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*public inline fun BooleanArray.getOrNull(index: Int): Boolean? {

return if (index >= 0 && index <= lastIndex) get(index) else null

*/

* Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
```kotlin
BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nCharArray.indexOfFirst(predicate: (Char) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nCharArray.indexOfFirst(predicate: (Char) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nIntArray.indexOfFirst(predicate: (Int) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nShortArray.indexOfFirst(predicate: (Short) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int { for (index in indices) { if (predicate(this[index])) { return index } } return -1 }\n
*\n//\n\n\nIntArray.indexOfLast(predicate: (Int) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nIntArray.indexOfLast(predicate: (Int) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nIntArray.indexOfLast(predicate: (Int) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n\nCharArray.indexOfLast(predicate: (Char) -> Boolean): Int { for (index in indices.reversed()) { if (predicate(this[index])) { return index } return -1 } }\n
*\n//\n\n```
element matching the given \[\text{predicate}\], or \`null\' if no such element was found.

```kotlin
DoubleArray.lastOrNull predicate: (Double) -> Boolean: Double? {
  for (index in this.indices.reversed()) {
    // Returns the last element matching the given predicate, or 'null' if no such element was found.
  }
}

BooleanArray.lastOrNull predicate: (Boolean) -> Boolean: Boolean? {
  for (index in this.indices.reversed()) {
    // Returns the last element matching the given predicate, or 'null' if no such element was found.
  }
}

CharArray.lastOrNull predicate: (Char) -> Boolean: Char? {
  for (index in this.indices.reversed()) {
    // Returns the last element matching the given predicate, or 'null' if no such element was found.
  }
}

ByteArray.random(): Byte {
  return random(Random)
}

ShortArray.random(): Short {
  return random(Random)
}

IntArray.random(): Int {
  return random(Random)
}

LongArray.random(): Long {
  return random(Random)
}

FloatArray.random(): Float {
  return random(Random)
}

DoubleArray.random(): Double {
  return random(Random)
}

BooleanArray.random(): Boolean {
  return random(Random)
}

CharArray.random(): Char {
  return random(Random)
}
```

Open Source Used In DNAC Platform Arica 11976
NoSuchElementException("Array is empty.")
        return get(random.nextInt(size))
    }\n\n    * Returns a random element from this array using the specified source of randomness.
    * @param random random source
    * @return a random element
    * @throws NoSuchElementException if this array is empty.
    */
    public fun DoubleArray.random(random: Random): Double {
        if (isEmpty())
            throw NoSuchElementException("Array is empty.")
        return get(random.nextInt(size))
    }\n\n    * Returns an element from this array matching the given [predicate], or throws an exception if there is no or more than one matching element.
    * @param predicate predicate to apply to each element
    * @return an element
    * @throws NoSuchElementException if this array is empty.
    * @throws IllegalArgumentException if there is no or more than one matching element.
    */
    public fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
        var single: Double? = null
        var found = false
        for (element in this) {
            if (predicate(element)) {
                if (found) throw IllegalArgumentException("Array contains more than one matching element.")
                single =
            }
        }
        if (single == null) throw NoSuchElementException("Array is empty.")
        return single
    }
element
    found = true
    } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as $T
  * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
  */
  @public inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {
    var single: Byte? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Byte
  *
  */
  @public inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {
    var single: Short? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Short
  *
  */
  @public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @ Suppress("UNCHECKED_CAST")
    return single as Int
  *
  */
  @public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Long
  *
  */
  @public inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Float
  *
  */
  @public inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Double
  *
  */
  @public inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Boolean
  *
  */
  @public inline fun CharArray.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
      if (predicate(element)) {
        if (found) throw IllegalArgumentException("Array contains more than one matching element.")
        single = element
        found = true
      } |n |n if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    } @Suppress("UNCHECKED_CAST")
    return single as Char
  *
element
    found = true
    )
} else if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")

@Suppress("UNCHECKED_CAST")
    return single as Char
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun <T> Array<out T>.singleOrNull(): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the given predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Char
}
BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun CharArray.singleOrNull(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> Array<out T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ByteArray.drop(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ShortArray.drop(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun IntArray.drop(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun LongArray.drop(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.drop(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.drop(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.drop(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun CharArray.dropLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ByteArray.dropLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ShortArray.dropLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun IntArray.dropLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun LongArray.dropLast(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.dropLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.dropLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.dropLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

Open Source Used In DNAC Platform Arica 11980
IllegalArgumentException if \( n \) is negative.

```kotlin
public fun ByteArray.dropLast(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun ShortArray.dropLast(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun IntArray.dropLast(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun LongArray.dropLast(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun FloatArray.dropLast(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun DoubleArray.dropLast(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun BooleanArray.dropLast(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

public fun CharArray.dropLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0)).toByteArray() * throw
}

```
Open Source Used In DNAC Platform Arica 11982

samples.collections.Collections.Transformations.drop

```kotlin
/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 */
public inline fun LongArray.dropLastWhile(predicate: (Long) -> Boolean): List<Long> {
    var yielding = false
    val list = ArrayList<Long>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 */
public inline fun FloatArray.dropLastWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 */
public inline fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 */
public inline fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    var yielding = false
    val list = ArrayList<Boolean>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 */
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {
    var yielding = false
    val list = ArrayList<Char>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {
    var yielding = false
    val list = ArrayList<Byte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {
    var yielding = false
    val list = ArrayList<Short>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {
    var yielding = false
    val list = ArrayList<Int>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {
    var yielding = false
    val list = ArrayList<Long>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```
the index of an element and the element itself
* and returns the result of predicate evaluation on the element.
*

//public inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float>
{
    return
filterIndexedTo(ArrayList<Float>(), predicate)\n\n\n* Returns a list containing only elements matching the
given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and
returns the result of predicate evaluation on the element.\n*

//public inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double>
{
    return
filterIndexedTo(ArrayList<Double>(), predicate)\n\n\n* Returns a list containing only elements matching the
given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and
returns the result of predicate evaluation on the element.\n*

//public inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean>
{
    return
filterIndexedTo(ArrayList<Boolean>(), predicate)\n\n\n* Returns a list containing only elements matching the
given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and
returns the result of predicate evaluation on the element.\n*

//public inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char>
{
    return
filterIndexedTo(ArrayList<Char>(), predicate)\n\n\n* Returns a list containing only elements matching the
given [predicate].\n* @param [predicate] function that takes the index of an element and the element itself\n* and
returns the result of predicate evaluation on the element.\n*
Open Source Used In DNAC Platform Arica 11985

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 @param [predicate] function that takes the index of an element and the element itself
 and returns the result of predicate evaluation on the element.
 @return destination
 */
public inline fun <C : MutableCollection<in Boolean>>
CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {

    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 @param [predicate] function that takes the index of an element and the element itself
 and returns the result of predicate evaluation on the element.
 */
public inline fun <C : MutableCollection<in Char>>
CharArray.filterNotTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {
    forEachIndexed { index, element ->
        if (!predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 */
public inline fun <reified R> Array<*>.filterIsInstance(): List<R> {
    return filterIsInstanceTo(ArrayList<R>(), predicate = { index, element -> element is R })
}

/**
 * Appends all elements that are instances of specified type parameter R to the given [destination].
 */
public inline fun <reified R, C : MutableCollection<in R>>
Array<*>.filterIsInstanceTo(destination: C): C {
    forEach { element ->
        if (element is R) destination.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {
    return filterNotTo(ArrayList<T>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {
    return filterNotTo(ArrayList<Byte>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {
    return filterNotTo(ArrayList<Short>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {
    return filterNotTo(ArrayList<Int>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {
    return filterNotTo(ArrayList<Long>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {
    return filterNotTo(ArrayList<Float>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double> {
    return filterNotTo(ArrayList<Double>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean): List<Boolean> {
    return filterNotTo(ArrayList<Boolean>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public inline fun CharArray.filterNot(predicate: (Char) -> Boolean): List<Char> {
    return filterNotTo(ArrayList<Char>(), predicate)
}

/**
 * Returns a list containing all elements that are not `null`.
 */
public fun <T : Any> Array<out T?>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())
}

/**
 * Appends all elements that are not `null` to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Array<out T?>.filterNotNullTo(destination: C): C {
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <T, C : MutableCollection<in T>>
Array<out T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}
MutableCollection<Int>> IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableCollection<Int>> Long>> LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableCollection<Int>> Float>> FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableCollection<Int>> Double>> DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableCollection<Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableCollection<Char>> CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <T, C : MutableCollection<in T>> Array<out T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Byte>> ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Byte>> ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <T, C : MutableList<in T>> Array<out T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Byte>> ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Short>> ShortArray.filterTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Int>> IntArray.filterTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Long>> LongArray.filterTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Float>> FloatArray.filterTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Double>> DoubleArray.filterTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Boolean>> BooleanArray.filterTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"%n */public inline fun <C : MutableList<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}"

/**
 * Returns a list containing elements at indices in the specified 
 * [indices] range.
 */
public fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified 
 * [indices] range.
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified 
 * [indices] range.
 */
public fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified 
 * [indices] range.
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
containing elements at indices in the specified [indices] range.

/**
 * Returns a list containing elements at specified [indices].
 */
public fun IntArray.sliceArray(indices: Collection<Int>): Array<Int> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun BooleanArray.sliceArray(indices: Collection<Int>): Array<Boolean> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun CharArray.sliceArray(indices: Collection<Int>): Array<Char> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun <T> Array<out T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun ByteArray.sliceArray(indices: Collection<Int>): Array<Byte> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun FloatArray.sliceArray(indices: Collection<Int>): Array<Float> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun DoubleArray.sliceArray(indices: Collection<Int>): Array<Double> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun BooleanArray.sliceArray(indices: Collection<Int>): Array<Boolean> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun CharArray.sliceArray(indices: Collection<Int>): Array<Char> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns a list containing elements of this array at specified [indices].
 */
public fun <T> Array<out T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

Array<T>.sliceArray(indices: Collection<Int>): Array<T> {
    val result = arrayOfNulls(this, indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}
result

* Returns an array containing elements of this array at specified [indices].

*public fun

ByteArray.sliceArray(indices: Collection<Int>): ByteArray {
    val result = ByteArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
public fun ShortArray.sliceArray(indices: Collection<Int>): ShortArray {
    val result = ShortArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

IntArray.sliceArray(indices: Collection<Int>): IntArray {
    val result = IntArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

LongArray.sliceArray(indices: Collection<Int>): LongArray {
    val result = LongArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

FloatArray.sliceArray(indices: Collection<Int>): FloatArray {
    val result = FloatArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

DoubleArray.sliceArray(indices: Collection<Int>): DoubleArray {
    val result = DoubleArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {
    val result = BooleanArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements of this array at specified [indices].

*public fun

CharArray.sliceArray(indices: Collection<Int>): CharArray {
    val result = CharArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

<T> Array<T>.sliceArray(indices: IntRange): Array<T> {
    if (indices.isEmpty()) return copyOfRange(0, 0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

ByteArray.sliceArray(indices: IntRange): ByteArray {
    if (indices.isEmpty()) return ByteArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

ShortArray.sliceArray(indices: IntRange): ShortArray {
    if (indices.isEmpty()) return ShortArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

IntArray.sliceArray(indices: IntRange): IntArray {
    if (indices.isEmpty()) return IntArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

LongArray.sliceArray(indices: IntRange): LongArray {
    if (indices.isEmpty()) return LongArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

DoubleArray.sliceArray(indices: IntRange): DoubleArray {
    if (indices.isEmpty()) return DoubleArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns an array containing elements at indices in the specified [indices] range.

*public fun

CharArray.sliceArray(indices: IntRange): CharArray {
    if (indices.isEmpty()) return CharArray(0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* Returns a list containing first [n] elements.

* @throws IllegalArgumentException if [n] is negative.
(Byte) -> Boolean): List<Byte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun ShortArray.takeLastWhile(predicate: Short -> Boolean): List<Short> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun IntArray.takeLastWhile(predicate: Int -> Boolean): List<Int> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun LongArray.takeLastWhile(predicate: Long -> Boolean): List<Long> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun FloatArray.takeLastWhile(predicate: Float -> Boolean): List<Float> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun DoubleArray.takeLastWhile(predicate: Double -> Boolean): List<Double> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun BooleanArray.takeLastWhile(predicate: Boolean -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun CharArray.takeLastWhile(predicate: Char -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun <T> Array<out T>.takeWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun ByteArray.takeWhile(predicate: (Byte) -> Boolean): List<Byte> {
    val list = ArrayList<Byte>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun ShortArray.takeWhile(predicate: (Short) -> Boolean): List<Short> {
    val list = ArrayList<Short>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun IntArray.takeWhile(predicate: (Int) -> Boolean): List<Int> {
    val list = ArrayList<Int>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun LongArray.takeWhile(predicate: (Long) -> Boolean): List<Long> {
    val list = ArrayList<Long>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun FloatArray.takeWhile(predicate: (Float) -> Boolean): List<Float> {
    val list = ArrayList<Float>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {
    val list = ArrayList<Double>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun BooleanArray.takeWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    val list = ArrayList<Boolean>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Transformations.take
 */
public inline fun CharArray.takeWhile(predicate: (Char) -> Boolean): List<Character> {
    val list = ArrayList<Character>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}
elements satisfying the given [predicate].\n
*\n
*/

//

public inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {
    val list = ArrayList<Double>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun DoubleArray.takeWhile(predicate: (Double) -> Boolean): List<Double> {
    val list = ArrayList<Double>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun <T> Array<T>.takeWhile(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun BooleanArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun ByteArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun IntArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun LongArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun ShortArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun CharArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun DoubleArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun FloatArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun ByteArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/**
 * Reverses elements in the array in-place.
 */
public fun <T> Array<out T>.reverse(): List<T> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun LongArray.reversed(): List<Long> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun ShortArray.reversed(): List<Short> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun ByteArray.reversed(): List<Byte> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Array<out T>.reversed(): List<T> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun CharArray.reversed(): List<Char> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun DoubleArray.reversed(): List<Double> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun FloatArray.reversed(): List<Float> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun LongArray.reversed(): List<Long> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun IntArray.reversed(): List<Int> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun ByteArray.reversed(): List<Byte> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
Open Source Used In DNAC Platform Arica

IntArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun ShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 */
public fun ByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
        sortWith(reverseOrder())
    }

/**
 * Sorts elements in the array in-place descending according to their natural sort order.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public inline fun <T : Comparable<T>> Array<out T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareByDescending(selector))
}

/**
 * Sorts elements in the array in-place descending according to their natural sort order of the value returned by specified [selector] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public inline fun <T, R : Comparable<R>> Array<out T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1) sortWith(compareBy(selector))
}

/**
 * Sorts elements in the array in-place according to natural sort order of the value returned by specified [selector] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public fun <T> Array<T>.reversedArray(): Array<T> {
    if (isEmpty()) return this
    val result = arrayOfNulls(this, size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

FloatArray.reversed(): List<Float> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

DoubleArray.reversed(): List<Double> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

BooleanArray.reversed(): List<Boolean> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

CharArray.reversedArray(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversedArray(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): List<Char> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

BooleanArray.reversed(): List<Boolean> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversedArray(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

CharArray.reversed(): CharArray {
    if (isEmpty()) return this
    val result = CharArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

ByteArray.reversed(): ByteArray {
    if (isEmpty()) return this
    val result = ByteArray(size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}
in the array in-place descending according to their natural sort order.

```kotlin
public fun LongArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun FloatArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun IntArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun ShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun ByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<T>.sortedArrayDescending(): List<T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T> List<T>.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T> List<T>.sortedArray(): List<T> {
    return toTypedArray().apply { sort() }.asList()
}
```

```kotlin
public fun <T> List<T>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Byte>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Int>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Float>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Long>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Short>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Char>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Double>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T> List<Byte>.sorted(): List<T> {
    return sortedArray().asList()
}
```

```kotlin
public fun ByteArray.sortedArray(): List<Byte> {
    return toTypedArray().apply { sort() }.asList()
}
```

```kotlin
public fun CharArray.sortedArray(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun IntArray.sortedArray(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun ShortArray.sortedArray(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun LongArray.sortedArray(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun FloatArray.sortedArray(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun DoubleArray.sortedArray(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun ByteArray.sortedArray(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun CharArray.sortedArray(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun DoubleArray.sortedArray(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun ShortArray.sortedArray(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> DoubleArray.sorted(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> ShortArray.sorted(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> CharArray.sorted(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> ByteArray.sorted(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> IntArray.sorted(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> LongArray.sorted(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> FloatArray.sorted(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sorted(): List<out T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sortedArrayDescending(): List<out T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sortedArray(): List<out T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sortedArrayHierarchical(): List<out T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> Array<out T>.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sorted(): List<out T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortedArrayDescending(): List<out T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortedArray(): List<out T> {
    return sortedArray().asList()
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortedHierarchical(): List<out T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortedArrayHierarchical(): List<out T> {
    return sort().apply { sort() }.asList()
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

```kotlin
public fun <T : Comparable<T>> List<out T>.sort(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```
their natural sort order.

```kotlin
public fun IntArray.sortedArrayDescending(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Returns a copy sorted in descending order
}

public fun LongArray.sortedArrayDescending(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Returns a copy sorted in descending order
}

public fun FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Returns a copy sorted in descending order
}

public fun DoubleArray.sortedArrayDescending(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Returns a copy sorted in descending order
}

public fun CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() } // Returns a copy sorted in descending order
}

public fun <T> Array<out T>.sortedArrayWith(comparator: Comparator<in T>): Array<out T> {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(comparator) } // Sorts elements according to the specified comparator
}

public inline fun <T, R : Comparable<R>> Array<out T>.sortedBy(crossinline selector: (T) -> R?): List<T> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> BooleanArray.sortedBy(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

public inline fun <R : Comparable<R>> CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareBy(selector)) // Sorts elements using the specified selector
}

```
Comparable<R>> ByteArray.sortedByDescending(crossinline selector: (Byte) -> R?): List<Byte> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> shortArray.sortedByDescending(crossinline selector: (Short) -> R?): List<Short> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> IntArray.sortedByDescending(crossinline selector: (Int) -> R?): List<Int> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> LongArray.sortedByDescending(crossinline selector: (Long) -> R?): List<Long> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public inline fun <R : Comparable<R>> CharArray.sortedByDescending(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareByDescending(selector)).unsorted *
} // Returns a list of all elements sorted descending according to natural sort order of the value returned by specified [selector] function.

public fun <T : Comparable<T>> Array<out T>.sortedDescending(): List<T> {
    return sortedArrayWith(reverseOrder()).unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun ByteArray.sortedDescending(): List<Byte> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun ShortArray.sortedDescending(): List<Short> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun IntArray.sortedDescending(): List<Int> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun LongArray.sortedDescending(): List<Long> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun FloatArray.sortedDescending(): List<Float> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun DoubleArray.sortedDescending(): List<Double> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

public fun CharArray.sortedDescending(): List<Char> {
    return copyOf().apply { sort() }.reversed().unsorted *
} // Returns a list of all elements sorted descending according to their natural sort order.

T>.sortedWith(comparator: Comparator<in T>): List<T> {
    return sortedArrayWith(comparator).unsorted *
} // Returns a list of all elements sorted according to the specified [comparator].

T>.sortedWith(comparator: Comparator<in Byte>): List<Byte> {
    return toTypedArray().apply { sortWith(comparator) }.unsorted *
} // Returns a list of all elements sorted according to the specified [comparator].

T>.sortedWith(comparator: Comparator<in Short>): List<Short> {
    return toTypedArray().apply { sortWith(comparator) }.unsorted *
} // Returns a list of all elements sorted according to the specified [comparator].
Comparator<in Int>): List<Int> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun LongArray.sortedWith(comparator: Comparator<in Long>): List<Long> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun FloatArray.sortedWith(comparator: Comparator<in Float>): List<Float> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun DoubleArray.sortedWith(comparator: Comparator<in Double>): List<Double> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun BooleanArray.sortedWith(comparator: Comparator<in Boolean>): List<Boolean> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun CharArray.sortedWith(comparator: Comparator<in Char>): List<Char> {
    return toTypedArray().apply { sortWith(comparator) }.asList()
}

/**
 * Returns a List that wraps the original array.
 */
public expect fun ByteArray.asList(): List<Byte>

/**
 * Returns a List that wraps the original array.
 */
public expect fun ShortArray.asList(): List<Short>

/**
 * Returns a List that wraps the original array.
 */
public expect fun IntArray.asList(): List<Int>

/**
 * Returns a List that wraps the original array.
 */
public expect fun LongArray.asList(): List<Long>

/**
 * Returns a List that wraps the original array.
 */
public expect fun ByteArray.asList(): List<Byte>

/**
 * Returns a List that wraps the original array.
 */
public expect fun FloatArray.asList(): List<Float>

/**
 * Returns a List that wraps the original array.
 */
public expect fun DoubleArray.asList(): List<Double>

/**
 * Returns a List that wraps the original array.
 */
public expect fun BooleanArray.asList(): List<Boolean>

/**
 * Returns a List that wraps the original array.
 */
public expect fun CharArray.asList(): List<Char>

/**
 * Returns a hash code based on the contents of this array as if it is a List.
 */
@SinceKotlin("1.1")
public expect fun <T> Array<out T>.contentDeepHashCode(): Int

/**
 * Returns a string representation of the contents of this array as if it is a List.
 */
@SinceKotlin("1.1")
public expect fun <T> Array<out T>.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ByteArray.contentEquals(other: ByteArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ShortArray.contentEquals(other: ShortArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun IntArray.contentEquals(other: IntArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun LongArray.contentEquals(other: LongArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun FloatArray.contentEquals(other: FloatArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun CharArray.contentEquals(other: CharArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ByteArray.contentDeepEquals(other: ByteArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ShortArray.contentDeepEquals(other: ShortArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun IntArray.contentDeepEquals(other: IntArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun LongArray.contentDeepEquals(other: LongArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun DoubleArray.contentDeepEquals(other: DoubleArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun FloatArray.contentDeepEquals(other: FloatArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun BooleanArray.contentDeepEquals(other: BooleanArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun CharArray.contentDeepEquals(other: CharArray): Boolean

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ByteArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun ShortArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun IntArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun LongArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun DoubleArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun FloatArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun BooleanArray.contentDeepToString(): String

/**
 * Returns `true` if the two specified arrays are structurally equal to one another.
 * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.1")
public expect infix fun CharArray.contentDeepToString(): String
Open Source Used In DNAC Platform Arica 11998

[equals][Any.equals] function.

For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun IntArray.contentEquals(other: IntArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun LongArray.contentEquals(other: LongArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun FloatArray.contentEquals(other: FloatArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun DoubleArray.contentEquals(other: DoubleArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun BooleanArray.contentEquals(other: BooleanArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun CharArray.contentEquals(other: CharArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun ByteArray.contentEquals(other: ByteArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun ShortArray.contentEquals(other: ShortArray):

Boolean?\n
* \n Returns ‘true’ if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.\n* \n* The elements are compared for equality with the [equals][Any.equals] function.\n* \n* For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n* \n}@SinceKotlin("1.1.1")\npublic expect infix fun IntArray.contentHashCode():

Int\n
* \n Returns a hash code based on the contents of this array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun ByteArray.contentHashCode():

Int?\n
* \n Returns a hash code based on the contents of this array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun ShortArray.contentHashCode():

Int?\n
* \n Returns a hash code based on the contents of this array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun IntArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun ByteArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun ShortArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun LongArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun FloatArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun DoubleArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun BooleanArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].\n* \n}@SinceKotlin("1.1.1")\npublic expect fun CharArray.contentToString():

String?\n
* \n Returns a string representation of the contents of the specified array as if it is [List].
@SinceKotlin("1.1")
public expect fun LongArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.1")
public expect fun FloatArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.1")
public expect fun DoubleArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.1")
public expect fun BooleanArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is [List].
 */
@SinceKotlin("1.1")
public expect fun CharArray.contentToString(): String

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T>

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
public expect fun EnumArray<out E>.copyInto(destination: EnumArray<E>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): EnumArray<E>
range of this array indices or when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
or when that index is out of the [destination] array indices range.
@throws IndexOutOfBoundsException or [IllegalStateException] when `startIndex > endIndex`.
specify the subrange so that it overlaps with the destination range.

```kotlin
@Throws(IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.)
@Throws(IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.)

@Return the [destination] array.

@SinceKotlin("1.3")
public expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):

```
public expect fun LongArray.copyOf(newSize: Int): LongArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedCopyOf
 */

public expect fun FloatArray.copyOf(newSize: Int): FloatArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedCopyOf
 */

public expect fun DoubleArray.copyOf(newSize: Int): DoubleArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with `false` values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `false` values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedCopyOf
 */

public expect fun BooleanArray.copyOf(newSize: Int): BooleanArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with null char (`\u0000`) values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\u0000`) values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedCopyOf
 */

public expect fun CharArray.copyOf(newSize: Int): CharArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with `null` values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.
 * @sample samples.collections.Arrays.CopyOfOperations.resizedCopyOf
 */

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOf(newSize: Int): Array<T?>

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T>

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
public expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T?>

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */

array which is a copy of the specified range of the original array.\n * \n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size` \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex])..\n * @public expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray\n * Returns a new array which is a copy of the specified range of the original array.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size` \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray\n * Returns a new array which is a copy of the specified range of the original array.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size` \n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex]).\n * @public expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit\n * Fills this array or its subrange with the specified [element] value.\n * @param fromIndex the start of the range (inclusive), must be in `0..array.size` \n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentRangeException if [fromIndex] is greater than [toIndex].\n * @public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit
array.

* @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

* @SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

* @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

*/
@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
public val <T: Array<out T>> Array<out T>.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val ShortArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val IntArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val LongArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val FloatArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val DoubleArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val BooleanArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
public val CharArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T: Array<out T>> Array<out T>.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.isEmpty(): Boolean {
    return size == 0
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns true if the array is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.isNotEmpty(): Boolean {
    return !isEmpty()
}

/**
 * Returns the last valid index for the array.
 */
public val <T> Array<out T>.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val ByteArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val ShortArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val IntArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val LongArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val FloatArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val DoubleArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val BooleanArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last valid index for the array.
 */
public val CharArray.lastIndex: Int
    get() = size - 1

/**
 * Open Source Used In DNAC Platform Arica 12004
the array.

```kotlin
public val ShortArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val IntArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val LongArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val FloatArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val DoubleArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val BooleanArray.lastIndex: Int
    get() = size - 1

/**
 * Returns the last index for the array.
 */
public val CharArray.lastIndex: Int
    get() = size - 1

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<T>.plus(element: T): Array<T>

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun ByteArray.plus(element: Byte): ByteArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun ShortArray.plus(element: Short): ShortArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun IntArray.plus(element: Int): IntArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun LongArray.plus(element: Long): LongArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun FloatArray.plus(element: Float): FloatArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun DoubleArray.plus(element: Double): DoubleArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun BooleanArray.plus(element: Boolean): BooleanArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun CharArray.plus(element: Char): CharArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray

@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun ShortArray.plus(elements: Collection<Short>): ShortArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun IntArray.plus(elements: Collection<Int>): IntArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun LongArray.plus(elements: Collection<Long>): LongArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun FloatArray.plus(elements: Collection<Float>): FloatArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray

@Suppress("NO.ACTUAL_FOR_EXPECT")
public expect operator fun CharArray.plus(elements: Collection<Char>): CharArray
```
given \[\text{elements}\] array.

```kotlin
public expect operator fun IntArray.plus(elements: IntArray): IntArray
```

Returns an array containing all elements of the original array and then all elements of the given \[\text{elements}\] array.

```kotlin
public expect operator fun LongArray.plus(elements: LongArray): LongArray
```

Returns an array containing all elements of the original array and then all elements of the given \[\text{elements}\] array.

```kotlin
public expect operator fun FloatArray.plus(elements: FloatArray): FloatArray
```

Returns an array containing all elements of the original array and then all elements of the given \[\text{elements}\] array.

```kotlin
public expect operator fun CharArray.plus(elements: CharArray): CharArray
```

Returns an array containing all elements of the original array and then the given \[\text{elements}\] array.

```kotlin
public expect operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray
```

Returns an array containing all elements of the original array and then all elements of the given \[\text{elements}\] array.

```kotlin
public expect operator fun CharArray.plus(elements: CharArray): CharArray
```

Returns an array containing all elements of the original array and then the given \[\text{elements}\] array.

```kotlin
public expect operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray
```

Returns an array containing all elements of the original array and then all elements of the given \[\text{elements}\] array.

```kotlin
public expect operator fun ByteArray.plusElement(element: Byte): ByteArray
```

Sorts the array in-place.

```kotlin
public expect fun <T> Array<T>.sort(): Unit
```

Sorts the array in-place.

```kotlin
public expect fun ByteArray.sort(): Unit
```

Sorts the array in-place.

```kotlin
public expect fun ShortArray.sort(): Unit
```

Sorts the array in-place.

```kotlin
public expect fun IntArray.sort(): Unit
```

Sorts the array in-place.

```kotlin
public expect fun CharArray.sort(): Unit
```

Sorts the array in-place.

```kotlin
public expect fun BooleanArray.sort(): Unit
```

Sorts the array in-place.

```kotlin
public fun Array<out Boolean>.toBooleanArray(): BooleanArray
```

Returns an array of Boolean containing all of the elements of this generic array.

```kotlin
public fun Array<out Byte>.toByteArray(): ByteArray
```

Returns an array of Byte containing all of the elements of this generic array.

```kotlin
public fun Array<out Char>.toCharArray(): CharArray
```

Returns an array of Char containing all of the elements of this generic array.

```kotlin
public fun Array<out Double>.toDoubleArray(): DoubleArray
```

Returns an array of Double containing all of the elements of this generic array.

```kotlin
public fun Array<out Float>.toFloatArray(): FloatArray
```

Returns an array of Float containing all of the elements of this generic array.

```kotlin
public fun Array<out Int>.toIntArray(): IntArray
```

Returns an array of Int containing all of the elements of this generic array.

```kotlin
public fun ByteArray.toTypedArray(): Array<Byte>
```

Returns a typed object array containing all of the elements of this primitive array.

```kotlin
public fun ShortArray.toTypedArray(): Array<Short>
```

Returns a typed object array containing all of the elements of this primitive array.

```kotlin
public fun IntArray.toTypedArray(): Array<Int>
```

Returns a typed object array containing all of the elements of this primitive array.
of the elements of this primitive array.

```kotlin
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray.toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray toTypedArray(): Array<Char>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray toTypedArray(): Array<Byte>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray toTypedArray(): Array<Short>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray toTypedArray(): Array<Int>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray toTypedArray(): Array<Long>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray toTypedArray(): Array<Float>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray toTypedArray(): Array<Double>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun BooleanArray toTypedArray(): Array<Boolean>
```

```kotlin
/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray toTypedArray(): Array<Char>
```
elements of the given array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.

The returned map preserves the entry iteration order of the original array.
returned from [keySelector] function applied to each element. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

```kotlin
public inline fun <K> CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector)
}
```

```kotlin
public inline fun <T> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K> BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Boolean>(capacity), keySelector)
}
```
for (element in this) {
    destination.put(keySelector(element), element)
} return destination

* Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function applied to each element of the given array * and value is the element itself. If any two
elements would have the same key returned by [keySelector] the last one gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.

.Populates and returns the [destination] mutable map with key-value pairs. * where key is provided by the
[keySelector] function and * and value is provided by the [valueTransform] function applied to elements of
the given array. * * If any two elements would have the same key returned by [keySelector] the last one
gets added to the map.
map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        destination.put(keySelector(element),
        valueTransform(element))
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any two pairs would have the same key the last one gets added to the map. 

```kotlin
public inline fun <T, K, V : MutableMap<in K, in V>> Array<out T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateTo(destination: M, transform: (Byte) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. 

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

```kotlin
BooleanArray.associateTo(destination: M, transform: (Boolean) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```

Open Source Used In DNAC Platform Arica 12012
* If any of two pairs would have the same key the last one gets added to the map.

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */
public fun <T, C : MutableCollection<in T>> Array<out T>.toCollection(destination: C) = C { for (item in this) {
    destination.add(item)
} return destination
}

/**
 * Returns a [HashSet] of all elements.
 */
public fun <T> Array<out T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

/**
 * Returns a [List] containing all elements.
 */
public fun <T> Array<out T>.toList(): List<T> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun FloatArray.toSet(): Set<Float> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun LongArray.toSet(): Set<Long> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun IntArray.toSet(): Set<Int> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun ShortArray.toSet(): Set<Short> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun ByteArray.toSet(): Set<Byte> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> this.toMutableSet()
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun <T> Array<out T>.toSet(): Set<T> {
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    val list = ArrayList<Byte>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun BooleanArray.toMutableList(): MutableList<Boolean> {
    val list = ArrayList<Boolean>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun DoubleArray.toMutableList(): MutableList<Double> {
    val list = ArrayList<Double>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun FloatArray.toMutableList(): MutableList<Float> {
    val list = ArrayList<Float>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun LongArray.toMutableList(): MutableList<Long> {
    val list = ArrayList<Long>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun IntArray.toMutableList(): MutableList<Int> {
    val list = ArrayList<Int>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun ShortArray.toMutableList(): MutableList<Short> {
    val list = ArrayList<Short>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toMutableList(): MutableList<Byte> {
    val list = ArrayList<Byte>(size)
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun <T> Array<out T>.toMutableList(): MutableList<T> {
    return
    for (item in this)
        list.add(item)
    return list
}

/**
 * Returns a [MutableList] filled with all elements of this array.
 */
public fun ByteArray.toList(): List<Byte> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun BooleanArray.toList(): List<Boolean> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun DoubleArray.toList(): List<Double> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun FloatArray.toList(): List<Float> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun LongArray.toList(): List<Long> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun IntArray.toList(): List<Int> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun ShortArray.toList(): List<Short> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}

/**
 * Returns a [List] containing all elements.
 */
public fun ByteArray.toList(): List<Byte> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}
toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}

/**
 * Returns a \[Set\] of all elements.
 * \n * The returned set preserves the element iteration order of the original array.
 * */
public fun BooleanArray.toSet(): Set<Boolean> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))
    }
}

/**
 * Returns a \[Set\] of all elements.
 * \n * The returned set preserves the element iteration order of the original array.
 * */
public fun CharArray.toSet(): Set<Char> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else -> toCollection(LinkedHashSet<Char>(mapCapacity(size)))
    }
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <T, R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of \[transform\] function being invoked on each element of original array.
 * */
public fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].
 * */
public fun <T, R, C : MutableCollection<in R>> Array<out T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].
 * */
public fun <R, C : MutableCollection<in R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].
 * */
public fun <R, C : MutableCollection<in R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].
 * */
public fun <R, C : MutableCollection<in R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of \[transform\] function being invoked on each element of original array, to the given \[destination\].
 * */
public fun <R, C : MutableCollection<in R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
list = transform(element)
destination.addAll(list)
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <T, K> Array<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <K> ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>> {
    return groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <K> ShortArray.groupBy(keySelector: (Short) -> K): Map<K, List<Short>> {
    return groupByTo(LinkedHashMap<K, MutableList<Short>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <K> IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>> {
    return groupByTo(LinkedHashMap<K, MutableList<Int>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
public inline fun <K> LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {
    return groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)
}
FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>>
{} return groupByTo(LinkedHashMap<K, MutableList<Float>>(), keySelector)
\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */

IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>>
{} return groupByTo(LinkedHashMap<K, MutableList<Int>>(), keySelector)

ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>>
{} return groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)

ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>>
{} return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)

CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>>
{} return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)

DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>>
{} return groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)

BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>>
{} return groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)

CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>>
{} return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)

MutableList.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>>
{} return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)

MutableArray.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>>
{} return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
with a list of corresponding values. The returned map preserves the entry iteration order of the keys produced from the original array.

```kotlin
public inline fun <K, V> LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```kotlin
public inline fun <K, V> CharArray.groupBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
```

```kotlin
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key){ ArrayList<T>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key){ ArrayList<Byte>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
/**
 * Groups elements of the original array by the key returned by the given 
 * [keySelector] function and puts to the 
 * [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key)
        list.add(element)
    }
    return destination
}
```

```kotlin
/**
 * Groups elements of the original array by the key returned by the given 
 * [keySelector] function and puts to the 
 * [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key)
        list.add(element)
    }
    return destination
}
```

```kotlin
/**
 * Groups elements of the original array by the key returned by the given 
 * [keySelector] function and puts to the 
 * [destination] map each group key associated with a list of corresponding elements.
 * @return The [destination] map.
 * @sample samples.collections.Collections.Transformations.groupBy
 */
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key)
        list.add(element)
    }
    return destination
}
```
fun <K, M : MutableMap<K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = 
        destination.getOrPut(key) {
            ArrayList<Short>()
        } add(element)
        return destination
    }

    Groups elements of the original array by the key returned by the given [keySelector] function. Each element and puts to the [destination] map each group key associated with a list of corresponding elements.

    @return The [destination] map.
}

public inline fun <K, M : MutableMap<K, MutableList<M>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = 
        destination.getOrPut(key) {
            ArrayList<Long>()
        } add(element)
        return destination
    }

    Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

    @return The [destination] map.
}

public inline fun <K, M : MutableMap<K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = 
        destination.getOrPut(key) {
            ArrayList<Boolean>()
        } add(element)
        return destination
    }

    Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

    @return The [destination] map.
}

public inline fun <K, M : MutableMap<K, MutableList<Char>>> CharArray.groupByTo(destination: M, keySelector: (Char) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = 
        destination.getOrPut(key) {
            ArrayList<Char>()
        } add(element)
        return destination
    }

    Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

    @return The [destination] map.
}

public inline fun <T, K, V, M : MutableMap<K, MutableList<V>>> Array<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = 
        destination.getOrPut(key) {
            ArrayList<V>()
        } add(valueTransform(element))
        return destination
    }

    Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding values.

    @return The [destination] map.
}
original array by the key returned by the given [keySelector] function applied to the element and puts to the [destination] map each group key associated with a list of corresponding values. 

* @return The [destination] map.

* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M { 
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M {
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}

*\n
inline fun <K, V, M : MutableMap<in K, MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {
  for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrElse(key) { ArrayList<V>() } 
    list.add(valueTransform(element)) 
  } 
  return destination
}
returns the result of the transform applied to the element.

`ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C { \n var index = 0 for (item in this) \n destination.add(transform(index++, item)) \n return destination}\n
Applies the given transform function to each element and its index in the original array and appends the results to the given destination.

`@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

`ShortArray.mapIndexedTo(destination: C, transform: (index: Int, Short) -> R): C { \n var index = 0 for (item in this) \n destination.add(transform(index++, item)) \n return destination}\n
Applies the given transform function to each element and its index in the original array and appends the results to the given destination.

`@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

`LongArray.mapIndexedTo(destination: C, transform: (index: Int, Long) -> R): C { \n var index = 0 for (item in this) \n destination.add(transform(index++, item)) \n return destination}\n
Applies the given transform function to each element and its index in the original array and appends the results to the given destination.

`@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

`IntArray.mapIndexedTo(destination: C, transform: (index: Int, Int) -> R): C { \n var index = 0 for (item in this) \n destination.add(transform(index++, item)) \n return destination}\n
Applies the given transform function to each element and its index in the original array and appends the results to the given destination.

`@param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

`ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C { \n var index = 0 for (item in this) \n destination.add(transform(index++, item)) \n return destination}\n
Applies the given transform function to each element and its index in the original array and appends the results to the given destination.
appends the results to the given [destination] \n *\n\npublic inline fun <R, C : MutableCollection<in R>>
ByteArray.mapTo(destination: C, transform: (Byte) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> ShortArray.mapTo(destination: C, transform: (Short) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> IntArray.mapTo(destination: C, transform: (Int) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> LongArray.mapTo(destination: C, transform: (Long) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> FloatArray.mapTo(destination: C, transform: (Float) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> DoubleArray.mapTo(destination: C, transform: (Double) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> BooleanArray.mapTo(destination: C, transform: (Boolean) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic inline fun <R, C : MutableCollection<in R>> CharArray.mapTo(destination: C, transform: (Char) -> R): C { \n    for (item in this)\n        destination.add(transform(item))\n    \n    return destination\n}\n\npublic fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}\n\npublic fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {
    return IndexingIterable { iterator() }
}\n\npublic fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {
    return IndexingIterable { iterator() }
}\n\npublic fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {
    return IndexingIterable { iterator() }
}\n\npublic fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}\n\npublic fun FloatArray.withIndex():
\npublic fun DoubleArray.withIndex():
\npublic fun BooleanArray.withIndex():
\npublic fun CharArray.withIndex():
only distinct elements from the given array.

The elements in the resulting list are in the same order as they were in the source array.

* `distinct()`: Returns a list containing only distinct elements from the given array.

* `distinctBy(selector: (T) -> K)`: Returns a list containing only distinct elements from the given array.

* `ByteArray.distinctBy(selector: (Byte) -> K)`: Returns a list containing only distinct elements from the given array.

* `ShortArray.distinctBy(selector: (Short) -> K)`: Returns a list containing only distinct elements from the given array.

* `IntArray.distinctBy(selector: (Int) -> K)`: Returns a list containing only distinct elements from the given array.

* `LongArray.distinctBy(selector: (Long) -> K)`: Returns a list containing only distinct elements from the given array.

* `FloatArray.distinctBy(selector: (Float) -> K)`: Returns a list containing only distinct elements from the given array.

* `DoubleArray.distinctBy(selector: (Double) -> K)`: Returns a list containing only distinct elements from the given array.

* `BooleanArray.distinctBy(selector: (Boolean) -> K)`: Returns a list containing only distinct elements from the given array.

* `CharArray.distinctBy(selector: (Char) -> K)`: Returns a list containing only distinct elements from the given array.

* `ByteArray.distinctBy(selector: (Byte) -> K)`: Returns a list containing only distinct elements from the given array.

* `ShortArray.distinctBy(selector: (Short) -> K)`: Returns a list containing only distinct elements from the given array.

* `IntArray.distinctBy(selector: (Int) -> K)`: Returns a list containing only distinct elements from the given array.

* `LongArray.distinctBy(selector: (Long) -> K)`: Returns a list containing only distinct elements from the given array.

* `FloatArray.distinctBy(selector: (Float) -> K)`: Returns a list containing only distinct elements from the given array.

* `DoubleArray.distinctBy(selector: (Double) -> K)`: Returns a list containing only distinct elements from the given array.

* `BooleanArray.distinctBy(selector: (Boolean) -> K)`: Returns a list containing only distinct elements from the given array.

* `CharArray.distinctBy(selector: (Char) -> K)`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.

* `List.toMutableSet().toList()`: Returns a list containing only distinct elements from the given array.
preserves the element iteration order of the original array. * To get a set containing all elements that are contained at least in one of these collections use [union]. * An public infix fun CharArray.intersect(other:
Iterable<Char>): Set<Char> {\n    val set = this.toMutableSet()\n    set.retainAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun <T> Array<out T>.subtract(other: Iterable<T>): Set<T> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Boolean> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun byteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun IntArray.subtract(other: Iterable<Int>): Set<Int> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun LongArray.subtract(other: Iterable<Long>): Set<Long> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Boolean> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun IntArray.subtract(other: Iterable<Int>): Set<Int> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun LongArray.subtract(other: Iterable<Long>): Set<Long> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Boolean> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun IntArray.subtract(other: Iterable<Int>): Set<Int> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun LongArray.subtract(other: Iterable<Long>): Set<Long> {\n    val set = this.toMutableSet()\n    set.removeAll(other)\n    return set\n}
* Returns a set containing all elements that are contained by this array and not contained by the specified collection. * An public infix fun
BooleanArray.union(other: Iterable<Boolean>): Set<Boolean> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns a set containing all distinct elements from both collections.
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.union(other: Iterable<Char>): Set<Char> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun ByteArray.all(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
isEmpty().
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <T> Array<out T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ByteArray.any(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ShortArray.any(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun IntArray.any(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun LongArray.any(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun FloatArray.any(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `false` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `false` if at least one element matches the given [predicate].
 * 
 * @sample
 * samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun CharArray.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.count(): Int {
    return size
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun <T> Array<out T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun IntArray.count(predicate: (Int) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}
/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <T, R> Array<out T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> IntArray.fold(initial: R, operation: (acc: R, Int) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
[operation] function that takes the index of an element, current accumulator value * and the element itself, and calculates the next accumulator value.

*/

public inline fun <R> ByteArray.foldRight(initial: R, operation: (Byte, acc: R) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 */

public inline fun <T, R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 */

public inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from left to right
 * to current accumulator value and each element with its index in the original array.
 */

public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 */

public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from left to right
 * to each element and current accumulator value.
 */

public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from left to right
 * to each element and current accumulator value.
 */

public inline fun <R> LongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 */

public inline fun <R> ShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from left to right
 * to each element and current accumulator value.
 */

public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value
 * starting with [initial] value and applying [operation] from right to left
 * to each element and current accumulator value.
 */

fun <R> Array<out T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) { /* O(n) */
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

fun <R> Array<out T>.foldLeft(initial: R, operation: (T, acc: R) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

fun <R> Array<out T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

fun <R> Array<out T>.foldIndexedRight(initial: R, operation: (T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

fun <R> Array<out T>.foldIndexedRight(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var index = lastIndex
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

fun <R> Array<out T>.foldIndexedLeft(initial: R, operation: (T, acc: R) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

fun <R> Array<out T>.foldIndexedLeft(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
= lastIndex

var accumulator = initial

while (index >= 0) {
    accumulator = operation(get(index--), accumulator)
}

return accumulator

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> IntArray.foldRightIndexed(initial: R, operation: (Int, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> ShortArray.foldRightIndexed(initial: R, operation: (Short, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> BooleanArray.foldRight(initial: R, operation: (Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> ByteArray.foldRightIndexed(initial: R, operation: (index: Int, Byte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> LongArray.foldRight(initial: R, operation: (Long, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> DoubleArray.foldRightIndexed(initial: R, operation: (Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> FloatArray.foldRightIndexed(initial: R, operation: (Float, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.
 */
public inline fun <R> IntArray.foldRightIndexed(initial: R, operation: (Int, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
operation: (index: Int, Int, acc: R) -> R
{\n    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
} // Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element in its original index and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself in
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element in its original index and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n*/
public inline fun <R> LongArray.foldRightIndexed(initial: R, operation: (Int, Int, Long, acc: R) -> R): R {\n    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
} // Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Accumulates value starting with [initial] value and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.\n* @param [operation] function that takes the index of an element, the element itself
* and current accumulator value, and calculates the next accumulator value.\n* /\n\n\n/**
* Performs the given [action] on each element, providing sequential index with the
* element.\n* /\n\n\n/**
* Performs the given [action] on each element, providing sequential index with the
* element.\n* /\n\n\n*/
public inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> ByteArray.forEach(action: (Byte) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> ShortArray.forEach(action: (Short) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> IntArray.forEach(action: (Int) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> LongArray.forEach(action: (Long) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> FloatArray.forEach(action: (Float) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> DoubleArray.forEach(action: (Double) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> BooleanArray.forEach(action: (Boolean) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> CharArray.forEach(action: (Char) -> Unit): Unit {\n    for (element in this) action(element)
} // Performs the given [action] on each element
*/
public inline fun <T> Array<out T>.forEachIndexed(action: (T, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> ByteArray.forEachIndexed(action: (Byte, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> ShortArray.forEachIndexed(action: (Short, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> IntArray.forEachIndexed(action: (Int, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> LongArray.forEachIndexed(action: (Long, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> FloatArray.forEachIndexed(action: (Float, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> DoubleArray.forEachIndexed(action: (Double, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> BooleanArray.forEachIndexed(action: (Boolean, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/
public inline fun <T> CharArray.forEachIndexed(action: (Char, Int) -> Unit): Unit {\n    for (index in this.indices) action(element, index)
} // Performs the given [action] on each element and its index
*/

inline fun <T> Array<out T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) 
        action(index++, item)
}

public fun LongArray.max(): Long? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
public fun IntArray.max(): Int? {
    if (isEmpty()) return
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
public fun ShortArray.max(): Short? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
public fun ByteArray.max(): Byte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Array<out Double>.max(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    if (max.isNaN()) return max
    for (i in 1..lastIndex) {
        val e = this[i]
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given action on each element, providing sequential index with the element.
 */
public inline function <T> Array<out T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public fun FloatArray.max(): Float? {
    if (isEmpty()) return null
    var max = this[0]
    if (max.isNaN()) return max
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public fun DoubleArray.max(): Double? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public fun CharArray.max(): Char? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <T, R : Comparable<R>> Array<out T>.maxBy(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <R : Comparable<R>> ByteArray.maxBy(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <R : Comparable<R>> ShortArray.maxBy(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <R : Comparable<R>> IntArray.maxBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <R : Comparable<R>> LongArray.maxBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public inline fun <R : Comparable<R>> FloatArray.maxBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
DoubleArray.maxBy(selector: (Double) -> R): Double? {\n    if (isEmpty()) return null
    var maxElem = this[0]
    lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = comparator.compare(minElem, e)
        if (v < 0) minElem = e
    }
    return minElem
}

/**
 * Returns the first element yielding the largest value of the given function or ‘null’ if there are no elements.
 */
public fun CharAr...
is `NaN` returns `NaN`.\n */
@SinceKotlin("1.1")
public fun Array<out Double>.min(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Array<out Float>.min(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun <T : Comparable<T>> Array<out T>.min(): T? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun ByteArray.min(): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun ShortArray.min(): Short? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun IntArray.min(): Int? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun LongArray.min(): Long? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun FloatArray.min(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun DoubleArray.min(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun CharArray.min(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    var e = this[0]
    if (min > e) min = e
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun minBy(selector: (T) -> R) = Array<out T>.minBy(selector)
IntArray.minBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    LongArray.minBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    FloatArray.minBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    DoubleArray.minBy(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    CharArray.minBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator]
 * or `null` if there are no elements.
 */
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```
samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samples.collections.UnitsWithPredicates<Byte>.noneWithPredicate(predicate: (Byte) -> Byte) { return true }

samples.collections.UnitsWithPredicates<Short>.noneWithPredicate(predicate: (Short) -> Short) { return true }

samples.collections.UnitsWithPredicates<Int>.noneWithPredicate(predicate: (Int) -> Int) { return true }

samples.collections.UnitsWithPredicates<Long>.noneWithPredicate(predicate: (Long) -> Long) { return true }

samples.collections.UnitsWithPredicates<Float>.noneWithPredicate(predicate: (Float) -> Float) { return true }

samples.collections.UnitsWithPredicates<Double>.noneWithPredicate(predicate: (Double) -> Double) { return true }

samples.collections.UnitsWithPredicates<Boolean>.noneWithPredicate(predicate: (Boolean) -> Boolean) { return true }

samples.collections.UnitsWithPredicates<Char>.noneWithPredicate(predicate: (Char) -> Char) { return true }

samplescollections.Unit

Open Source Used In DNAC Platform Arica

// Function to accumulate value starting with the first element and applying [operation] from left to right for each element.

fun List<*>.reduceIndexed(operation: (index: Int, acc: T, T) -> T): T {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right for each element.
 */
fun <S, T : S> Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right for each element.
 */
fun ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right for each element.
 */
fun IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right for each element.
 */
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right for each element.
 */
fun FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, acc, this[index])
    }
    return accumulator
}

// Additional functions for accumulations with index information and different types.
accumulator)
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun <S, T : S> Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 */
public inline fun FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float {
    var index =
lastIndex
    if (index &lt; 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index &gt;= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left using
 * a function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 */
public inline fun DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index &lt; 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index &gt;= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left using
 * a function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 */
public inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index &lt; 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index &gt;= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left using
 * a function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, the element itself, and
 * current accumulator value, and calculates the next accumulator value.
 */
public inline fun CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index &lt; 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index &gt;= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun <T> Array&lt;out T&gt;.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to
 * each element in the array.
 */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
(element in this) {\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun FloatArray.sumByDouble(selector: (Float) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun DoubleArray.sumByDouble(selector: (Double) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun FloatArray.sumByDouble(selector: (Float) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun DoubleArray.sumByDouble(selector: (Double) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
} \n/** \n* Returns the sum of all values produced by [selector] function applied to each element in the array.\n*/ \n\npublic inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {\nvar sum = Double = 0.0n \nfor (element in this) \n{\n    sum += selector(element) \n} \nreturn sum
}
elements for which \([\text{predicate}]\) yielded \(\text{false}\).\n*/

```kotlin
public inline fun LongArray.partition(predicate: (Long) -> Boolean): Pair<List<Long>, List<Long>> {
    val first = ArrayList<Long>()
    val second = ArrayList<Long>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
/**
 * Splits the original array into pair of lists,
 * where *first* list contains elements for which \([\text{predicate}]\) yielded \(\text{true}\),\n * while *second* list contains elements for which \([\text{predicate}]\) yielded \(\text{false}\).\n */

```kotlin
public inline fun FloatArray.partition(predicate: (Float) -> Boolean): Pair<List<Float>, List<Float>> {
    val first = ArrayList<Float>()
    val second = ArrayList<Float>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
/**
 * Splits the original array into pair of lists,
 * where *first* list contains elements for which \([\text{predicate}]\) yielded \(\text{true}\),\n * while *second* list contains elements for which \([\text{predicate}]\) yielded \(\text{false}\).\n */

```kotlin
public inline fun DoubleArray.partition(predicate: (Double) -> Boolean): Pair<List<Double>, List<Double>> {
    val first = ArrayList<Double>()
    val second = ArrayList<Double>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
/**
 * Splits the original array into pair of lists,
 * where *first* list contains elements for which \([\text{predicate}]\) yielded \(\text{true}\),\n * while *second* list contains elements for which \([\text{predicate}]\) yielded \(\text{false}\).\n */

```kotlin
public inline fun BooleanArray.partition(predicate: (Boolean) -> Boolean): Pair<List<Boolean>, List<Boolean>> {
    val first = ArrayList<Boolean>()
    val second = ArrayList<Boolean>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
/**
 * Splits the original array into pair of lists,
 * where *first* list contains elements for which \([\text{predicate}]\) yielded \(\text{true}\),\n * while *second* list contains elements for which \([\text{predicate}]\) yielded \(\text{false}\).\n */

```kotlin
public inline fun CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {
    val first = ArrayList<Char>()
    val second = ArrayList<Char>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <T, R> Array<out T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> ByteArray.zip(other: Array<out R>): List<Pair<Byte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> IntArray.zip(other: Array<out R>): List<Pair<Int, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> LongArray.zip(other: Array<out R>): List<Pair<Long, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
/**
 * Returns a list of pairs built from the elements of \(\text{this}\) array and the \[other\] array with the same\n * index.\n * The returned list has length of the shortest collection.\n */

```kotlin
public infix fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```
List<Pair<Double, R>> { t1, t2 -> t1 to t2 }. Returns a list of pairs built from the elements of ‘this’ array and the [other] array with the same index. * The returned list has length of the shortest collection. * * * @sample samples.collections.Operations.zipIterable

R>): List<Pair<Boolean, R>> 
{ t1, t2 -> t1 to t2 }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * The returned list has length of the shortest collection. * * * @sample samples.collections.Operations.zipIterable

CharArray.zip(other: Array<out R>): List<Pair<Char, R>> 
{ t1, t2 -> t1 to t2 }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * The returned list has length of the shortest collection. * * * @sample samples.collections.Operations.zipIterable

SignedCharArray.zip(other: Array<out R>, transform: CharTransform<R>): List<Pair<Char, R>> 
{ t1, t2 -> t1 to t2 }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * The returned list has length of the shortest collection. * * * @sample samples.collections.Operations.zipIterableWithTransform

ByteArray.zip(other: Array<out R>, transform: (Byte, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

ShortArray.zip(other: Array<out R>, transform: (Short, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

IntArray.zip(other: Array<out R>, transform: (Int, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

LongArray.zip(other: Array<out R>, transform: (Long, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

FloatArray.zip(other: Array<out R>, transform: (Float, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

DoubleArray.zip(other: Array<out R>, transform: (Double, R) -> V): List<V> 
{ val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } }. Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index. * Using the provided [transform] function applied to each pair of elements. * * * @sample samples.collections.Operations.zipIterableWithTransform

Open Source Used In DNAC Platform Arica 12048
with the same index\n * using the provided [transform] function applied to each pair of elements.\n * The returned
list has length of the shortest collection.\n * \n * @sample
samples.collections.Iterables.Operations.zipIterableWithTransform\n */
BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(\n        size\n    )\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}

CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {\n    val size = minOf(size, other.size)\n    val list = ArrayList<V>(\n        size\n    )\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}

Array<out T>.zip(other: Iterable<R>): List<Pair<T, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

IntArray.zip(other: Iterable<R>): List<Pair<Int, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

Array<out T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {\n    val arraySize = size\n    val list = ArrayList<V>(\n        minOf(other.collectionSizeOrDefault(10), arraySize)\n    )\n    var i = 0\n    for (element in other) {\n        list.add(transform(this[i], element))\n        i++\n    }\n    return list\n}
if (i >= arraySize) break
    list.add(transform(this[i++], element))
} return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}

/**
 * Returns a list of values built from the elements of ‘this’ array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    } return list
}
public inline fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> { val arraySize = this.size val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize)) var i = 0
for (element in other) {
    if (i >= arraySize) break
    list.add(transform(this[i++], element))
} return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
public inline fun <R, V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    } return list
}
ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <V> LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <V> BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <V> CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Appends the string from all the elements separated using [separator] and using the
 * given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of
 * [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string
 * (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <T, A : Appendable> Array<out T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
if (limit >= 0 && count > limit) buffer.append(truncated);
buffer.append(postfix);
return
buffer.append(prefix);
var count = 0;
for (element in this) {
    if (++count > 1)
        buffer.append(separator);
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.append(transform(element));
        else
            buffer.append(element.toString());
    } else break
}
buffer.append(separator)
if (limit < 0 || count <= limit) {
    if (transform != null)
        buffer.append(transform(element))
    else
        buffer.append(element.toString())
} else
    break
}
if (limit <= 0 || count > limit) buffer.append(transformed)
buffer.append(postfix)
return

/**
 * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <A : Appendable> A.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element)
        } else
            break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Array<in T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun ByteArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Byte) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun ShortArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun IntArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun LongArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun FloatArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun DoubleArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun BooleanArray.joinTo(buffer: Appendable, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): Appendable {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else
            break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit]* elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun CharArray.joinTo(buffer: Appendable, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): Appendable {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element)
        } else
            break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
CharSequence = ",", prefix: CharSequence = ",", postfix: CharSequence = ",", limit: Int = -1, truncated:
CharSequence = ",", transform: ((Int) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun LongArray.joinToString(separator: CharSequences = ",", prefix: CharSequences = ",", postfix: CharSequences = ",", limit: Int = -1, truncated:
CharSequences = ",", transform: ((Long) -> CharSequences)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun FloatArray.joinToString(separator: CharSequences = ",", prefix: CharSequences = ",", postfix: CharSequences = ",", limit: Int = -1, truncated:
CharSequences = ",", transform: ((Float) -> CharSequences)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun DoubleArray.joinToString(separator: CharSequences = ",", prefix: CharSequences = ",", postfix: CharSequences = ",", limit: Int = -1, truncated:
CharSequences = ",", transform: ((Double) -> CharSequences)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun BooleanArray.joinToString(separator: CharSequences = ",", prefix: CharSequences = ",", postfix: CharSequences = ",", limit: Int = -1, truncated:
CharSequences = ",", transform: ((Boolean) -> CharSequences)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun CharArray.joinToString(separator: CharSequences = ",", prefix: CharSequences = ",", postfix: CharSequences = ",", limit: Int = -1, truncated:
CharSequences = ",", transform: ((Char) -> CharSequences)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun ByteArray.asIterable(): Iterable<Byte> {
    return asIterable()}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun ShortArray.asIterable(): Iterable<Short> {
    return asIterable()}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun IntArray.asList(): Iterable<Int> {
    return asList()}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */
public fun LongArray.asList(): Iterable<Long> {
    return asList()}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to ",...").
 */

(isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun BooleanArray.asIterable(): Iterable<Boolean> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */
public fun CharArray.asIterable(): Iterable<Char> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun ByteArray.asSequence(): Sequence<Byte> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun ShortArray.asSequence(): Sequence<Short> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun IntArray.asSequence(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun FloatArray.asSequence(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
public fun DoubleArray.asSequence(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Array<out Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Array<out Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Array<out Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Array<out Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
Double = 0.0
var count: Int = 0
for (element in this) {
    sum += element
    ++count
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun Array<out Float>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun Array<out Double>.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ByteArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ShortArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun IntArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun LongArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun FloatArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun DoubleArray.average(): Double {
    var sum = 0.0
    var count = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Byte>.sum(): Int {
    var sum = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Short>.sum(): Int {
    var sum = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Int>.sum(): Int {
    var sum = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Long>.sum(): Long {
    var sum = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Float>.sum(): Float {
    var sum = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Double>.sum(): Double {
    var sum = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 */
license/LICENSE.txt file

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt

import kotlin.random.*

/**
 * Returns a random element from this range.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    try {
        return random.nextInt(this)
    } catch (e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Byte): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Byte): Boolean {
    return contains(value.toShort())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}

Open Source Used In DNAC Platform Arica 12058
floating point arguments has ambiguous semantics and is going to be removed.

```
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Int): Boolean { return value.toIntExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Double): Boolean { return contains(value.toDouble()) }

@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Double): Boolean { return value.toByteExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Double): Boolean { return value.toShortExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Double): Boolean { return contains(value.toFloat()) }

@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Float): Boolean { return value.toIntExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Float): Boolean { return value.toLongExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Float): Boolean { return value.toByteExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Float): Boolean { return value.toShortExactOrNull().let { if (it != null) contains(it) else false } }

@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Double): Boolean { return contains(value.toDouble()) }
```

floating point arguments has ambiguous semantics and is going to be removed.\n@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Int): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Long): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Short): Boolean {
    return contains(value.toLong())
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to ‘this’ value.\n
 */
public infix fun Int.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to ‘this’ value.\n
*/
public infix fun Long.downTo(to: Byte): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to ‘this’ value.\n
*/
public infix fun Byte.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(toToInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to ‘this’ value.\n
*/
public infix fun Short.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(toToInt(), to.toInt(), -1)
}
Char.downTo(to: Char): CharProgression {
    return CharProgression(fromClosedRange(this, to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Int.downTo(to: Int): IntProgression {
    return IntProgression(fromClosedRange(this, to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Int.downTo(to: Long): LongProgression {
    return LongProgression(fromClosedRange(this, to, toLong()), -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Byte.downTo(to: Int): IntProgression {
    return IntProgression(fromClosedRange(this.toInt(), to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Byte.downTo(to: Long): LongProgression {
    return LongProgression(fromClosedRange(this.toLong(), to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Short.downTo(to: Int): IntProgression {
    return IntProgression(fromClosedRange(this.toInt(), to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Short.downTo(to: Long): LongProgression {
    return LongProgression(fromClosedRange(this.toLong(), to, -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Int.downTo(to: Short): IntProgression {
    return IntProgression(fromClosedRange(this, to.toInt()), -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public infix fun Byte.downTo(to: Short): IntProgression {
    return IntProgression(fromClosedRange(this.toInt(), to.toInt()), -1L))
} *returns a progression from this value down to the specified [to] value with the step -1. The [to] value should be less than or equal to 'this' value. If the [to] value is greater than 'this' value the returned progression is empty.

*public fun IntProgression.reversed(): IntProgression {
    return IntProgression(fromClosedRange(last, first), -step)
} *returns a progression that goes over the same range in the opposite direction with the same step.

*public fun LongProgression.reversed(): LongProgression {
    return LongProgression(fromClosedRange(last, first), -step)
} *returns a progression that goes over the same range in the opposite direction with the same step.

*public fun CharProgression.reversed(): CharProgression {
    return CharProgression(fromClosedRange('z', 'a'))
} *returns a progression that goes over the same range in the opposite direction with the same step.
return CharProgression.fromClosedRange(last, first, -step))
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun IntProgression.step(step: Int): IntProgression {
    checkStepIsPositive(step > 0, step)
    return IntProgression.fromClosedRange(first, last, step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun LongProgression.step(step: Long): LongProgression {
    checkStepIsPositive(step > 0, step)
    return LongProgression.fromClosedRange(first, last, step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun CharProgression.step(step: Int): CharProgression {
    checkStepIsPositive(step > 0, step)
    return CharProgression.fromClosedRange(first, last, step)
}

internal fun Int.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null
}

internal fun Long.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null
}

internal fun Short.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null
}

internal fun Double.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toByte() else null
}

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toByte() else null
}

internal fun Int.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun Float.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null
}

internal fun Long.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toDouble()..Long.MAX_VALUE.toDouble()) this.toLong() else null
}

internal fun Float.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toFloat()..Long.MAX_VALUE.toFloat()) this.toLong() else null
}

internal fun Int.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt()) this.toShort() else null
}

internal fun Long.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toLong()..Short.MAX_VALUE.toLong()) this.toShort() else null
}

internal fun Double.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toDouble()..Short.MAX_VALUE.toDouble()) this.toShort() else null
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range
 * is empty.
 */
public infix fun IntProgression.until(to: Byte): IntRange {
    return this .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range
 * is empty.
 */
public infix fun LongProgression.until(to: Byte): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range
 * is empty.
 */
public infix fun CharProgression.until(to: Char): CharRange {
    return this .. (to - 1).toChar()
}
IntRange.EMPTY

    return this .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * @param range
 * @return the range
 */
public infix fun Byte.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this.toInt() .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * @param range
 * @return the range
 */
public infix fun Short.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this.toInt() .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * @param range
 * @return the range
 */
public infix fun Long.until(to: Int): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this.toInt() .. (to - 1).toInt()
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Byte.coerceAtLeast(minimumValue: Byte): Byte {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Short.coerceAtLeast(minimumValue: Short): Short {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Int.coerceAtLeast(minimumValue: Int): Int {
    return if (this < minimumValue) minimumValue else this
}
samples.comparisons.ComparableOps.coerceAtLeast
@public fun Long.coerceAtLeast(minimumValue: Long): Long {
    return if (this < minimumValue) minimumValue else this
}

public fun Float.coerceAtLeast(minimumValue: Float): Float {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * 
 * @return this value if it's greater than or equal to the [minimumValue]
 * or the [minimumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Double.coerceAtLeast(minimumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Byte.coerceAtMost(maximumValue: Byte): Byte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Short.coerceAtMost(maximumValue: Short): Short {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}

class Comparable {
    var value: Any

    init {
        this.value = 0
    }

    fun compare(other: Comparable): Int {
        return compare(other.value)
    }

    override fun toString(): String {
        return "Comparable(value = $value)"
    }

    fun asComparable(): Comparable {
        return Comparable(value = this.value)
    }
}

fun main() {
    val comparable = Comparable()
    println(comparable)
}

samples.comparisons.ComparableOps.coerceIn

```kotlin
public fun Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun Double.coerceIn(minimumValue: Double, maximumValue: Double): Double {
    if (minimumValue > maximumValue) throw
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
```

```kotlin
public fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw
    if (range.lessThanOrEquals(this, range.start) && !range.lessThanOrEquals(range.start, this)) return range.start
    if (range.lessThanOrEquals(range.endInclusive, this) && !range.lessThanOrEquals(this, range.endInclusive)) return range.endInclusive
    else return this
}
```
@return this value if it’s in the [range], or ‘range.start’ if this value is less than ‘range.start’, or ‘range.endInclusive’ if this value is greater than ‘range.endInclusive’.

samples.comparisons.ComparableOps.coerceInComparable

* /public fun <T : Comparable<T>>

T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    } else if (range.isEmpty()) throw IllegalArgumentException("Cannot coerect value to empty range: \${range}).

    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

* /public fun Int.coerceIn(range: ClosedRange<Int>): Int {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Int>(range)
    } else if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: \${range}).

    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

* /public fun Long.coerceIn(range: ClosedRange<Long>): Long {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<Long>(range)
    } else if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: \${range}).

    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

* /public fun UByte @PublishedApi internal constructor(@PublishedApi internal val data: Byte) : Comparable<UByte> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UByte can have.
         */
        public const val MIN_VALUE: UByte = UByte(0)

        /**
         * A constant holding the maximum value an instance of
         * UByte can have.
         */
        public const val MAX_VALUE: UByte = UByte(-1)

        /**
         * The number of bytes used to represent an instance of UByte in a binary form.
         */
        public const val SIZE_BYTES: Int = 1

        /**
         * The number of bits used to represent an instance of UByte in a binary form.
         */
        public const val SIZE_BITS: Int = 8
    }

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public override inline operator fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UShort): Int = this.toInt().compareTo(other.toInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)

    /**
     * Adds the other value to this value.
     * Returns the other value to this value.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UByte): UByte = this.toUInt().plus(other.toUInt())

    /**
     * Adds the other value to this value.
     * Returns the other value to this value.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UShort): UByte = this.toUInt().plus(other.toUInt())

    /**
     * Adds the other value to this value.
     * Returns the other value to this value.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UInt): UByte = this.toUInt().plus(other)

    /**
     * Adds the other value to this value.
     * Returns the other value to this value.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): UByte = this.toULong().plus(other)
public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other) /* Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt()) /* Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt()) /* Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt()) /* Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt()) /* Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt()) /* Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt()) /* Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong = this.toULong().div(other)
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt()) /* Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt()) /* Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): UByte = UByte(data.inc()) /* Increments this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): UByte = UByte(data.dec()) /* Increments this value. */
    @kotlin.internal.InlineOnly
    operator fun rangeTo(other: UByte): UIntRange = UIntRange(this.toUInt(), other.toUInt()) /* Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data) /* Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: UByte): UByte = UByte(this.data or other.data) /* Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UByte): UByte = UByte(this.data xor other.data) /* Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): UByte = UByte(data.inv()) /* Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): UInt = UByte(data.inv()) /* Inverts the bits in this value. */
    * Converts this [UByte] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): UShort = UByte(data.inv()) /* Inverts the bits in this value. */
    * Converts this [UByte] value to [Short]. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): Int = UByte(data.inv()) /* Inverts the bits in this value. */
    * Converts this [UByte] value to [Int]. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): Long = UByte(data.inv()) /* Inverts the bits in this value. */
    * Converts this [UByte] value to [Long]. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): Byte = UByte(data.inv()) /* Converts this [UByte] value to [Byte]. */
    * If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents the same numerical value as this `UByte`. Otherwise the result is negative. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): Short = UByte(data.inv()) /* Converts this [UByte] value to [Short]. */
    * The least significant 8 bits of the resulting `Short` value are the same as the bits of this `UByte` value, whereas the most significant 8 bits are filled with zeros. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): Int = UByte(data.inv()) /* Converts this [UByte] value to [Int]. */
    * The resulting `Int` value represents the same numerical value as this
The least significant 8 bits of the resulting `Int` value are the same as the bits of this `UByte`
value, whereas the most significant 24 bits are filled with zeros.

    /**
     * Converts this [UInt] value to [ULong].
     * The resulting `ULong` value represents the same numerical value as this `UInt`.
     * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte` value,
     * whereas the most significant 56 bits are filled with zeros.
     */
    @kotlin.internal.InlineOnly
    public inline fun toULong(): ULong = ULong(data.toLong() and 0xFF)

    /**
     * Converts this [UByte] value to [ULong].
     * The resulting `ULong` value represents the same numerical value as this `UByte`.
     * The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `UByte` value,
     * whereas the most significant 56 bits are filled with zeros.
     */
    @kotlin.internal.InlineOnly
    public inline fun toULong(): ULong = ULong(data.toLong() and 0xFF)

    /**
     * Converts this [UByte] value to [Float].
     * The resulting `Float` value represents the same numerical value as this `UByte`.
     * If this value is positive and less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
     * the same numerical value as this `Short`.
     * The resulting `UByte` value is represented by the least significant 8 bits of this `Short` value.
     */
    @kotlin.internal.InlineOnly
    public inline fun toFloat(): Float = this.toInt().toFloat()
* A constant holding the minimum value an instance of UInt can have.

```kotlin
public const val MIN_VALUE: UInt = UInt(0)
```

* A constant holding the maximum value an instance of UInt can have.

```kotlin
public const val MAX_VALUE: UInt = UInt(-1)
```

* The number of bytes used to represent an instance of UInt in a binary form.

```kotlin
public const val SIZE_BYTES: Int = 4
```

* The number of bits used to represent an instance of UInt in a binary form.

```kotlin
public const val SIZE_BITS: Int = 32
```

* Compares this value with the specified value for order. Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.

```kotlin
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())
public inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)
public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)
```

* Compares this value with the specified value for order. Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.

```kotlin
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())
public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toUInt())
public inline operator fun compareTo(other: ULong): Int = this.compareTo(other.toULong())
```

* Adds the other value to this value.

```kotlin
public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())
public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt())
public inline operator fun plus(other: UInt): UInt = UInt(this.data.plus(other.data))
public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)
```

* Subtracts the other value from this value.

```kotlin
public inline operator fun minus(other: UByte): UInt = this.minus(other.toUInt())
public inline operator fun minus(other: UShort): UInt = this.minus(other.toUInt())
public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data))
public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
```

* Multiplies this value by the other value.

```kotlin
public inline operator fun times(other: UByte): UInt = this.times(other.toUInt())
public inline operator fun times(other: UShort): UInt = this.times(other.toUInt())
public inline operator fun times(other: UInt): UInt = UInt(this.data.times(other.data))
public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
```

* Divides this value by the other value.

```kotlin
public inline operator fun div(other: UByte): UInt = this.div(other.toUInt())
public inline operator fun div(other: UShort): UInt = this.div(other.toUInt())
public inline operator fun div(other: UInt): UInt = uintDivide(this, other)
public inline operator fun div(other: ULong): ULong = this.toULong().div(other)
```

* Calculates the remainder of dividing this value by the other value.

```kotlin
public inline operator fun rem(other: UByte): UInt = this.rem(other.toUInt())
public inline operator fun rem(other: UShort): UInt = this.rem(other.toUInt())
public inline operator fun rem(other: UInt): UInt = this.rem(other.toUInt())
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
```

---

Open Source Used in DNAC Platform Arica 12069
Open Source Used In DNAC Platform Arica 12070

```kotlin
this.UInt.value to Double.
     *
     * The resulting `Double` value represents the same numerical value as this `UInt` value.
     */

@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this `UInt` value to `Float`.
 *
 * The resulting `Float` value represents the same numerical value as this `UInt` value.
 *
 * The least significant 32 bits of the resulting `Float` value are the same as the bits of this `UInt` value; whereas the most significant 32 bits are filled with zeros.
 */

@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF_FFFF)

/**
 * Converts this `UInt` value to `ULong`.
 *
 * The resulting `ULong` value represents the same numerical value as this `UInt`.
 *
 * The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `UInt` value; whereas the most significant 32 bits are filled with zeros.
 */

@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = this

/**
 * Converts this `UInt` value to `Int`.
 *
 * If this value is less than or equals to `Int.MAX_VALUE`, the resulting `Int` value represents the same numerical value as this `UInt` value. Otherwise the result is negative.
 */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/**
 * Converts this `UInt` value to `Short`.
 *
 * If this value is less than or equals to `Short.MAX_VALUE`, the resulting `Short` value represents the same numerical value as this `UInt` value.
 *
 * The least significant 32 bits of the resulting `Short` value are the same as the bits of this `UInt` value; whereas the most significant 32 bits are filled with zeros.
 */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/**
 * Converts this `UInt` value to `Byte`.
 *
 * If this value is less than or equals to `Byte.MAX_VALUE`, the resulting `Byte` value represents the same numerical value as this `UInt` value. Otherwise the result is negative.
 */

@kotlin.internal.InlineOnly
public inline fun inv(): UInt = UInt(data.inv())

/**
 * Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other) /* Increments this value. */

@kotlin.internal.InlineOnly
public inline operator fun inc(): UInt = UInt(data.inc()) /* Decrements this value. */

@kotlin.internal.InlineOnly
public inline operator fun dec(): UInt = UInt(data.dec()) /* Creates a range from this value to the specified [other] value. */

@kotlin.internal.InlineOnly
public inline operator fun shl(bitCount: Int): UInt = UInt(data shl bitCount) /* Performs a bitwise AND operation between the two values. */

@kotlin.internal.InlineOnly
public inline operator fun shr(bitCount: Int): UInt = UInt(data ushr bitCount) /* Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = uintRemainder(this, other) /* Converts this [UInt] value to [Byte]. */

@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UInt): UIntRange = UIntRange(this, other) /* Creates a range from this value to the specified [other] value. */

@kotlin.internal.InlineOnly
public inline operator fun dec(): UInt = UInt(data.dec()) /* Decrements this value. */

@kotlin.internal.InlineOnly
public inline operator fun inc(): UInt = UInt(data.inc()) /* Increments this value. */

@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UInt): UIntRange = UIntRange(this, other) /* Creates a range from this value to the specified [other] value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = uintRemainder(this, other) /* Calculates the remainder of dividing this value by the other value. */
```
Open Source Used In DNAC Platform Arica 12071

```kotlin
public override fun toString(): String = toLong().toString()

/**
 * Converts this [Byte] value to [UInt].
 *
 * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte` value.
 * The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `Byte` value, whereas the most significant 24 bits are filled with the sign bit of this value.
 */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = uintToDouble(data)

/**
 * Converts this [Byte] value to [UInt].
 *
 * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte` value.
 * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Byte` value, whereas the most significant 16 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toUInt(): UInt = UInt(this.toInt())

/**
 * Converts this [Short] value to [UInt].
 *
 * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Short` value.
 * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value, whereas the most significant 16 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toUInt(): UInt = UInt(this.toInt())

/**
 * Converts this [Int] value to [UInt].
 *
 * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Int` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toUInt(): UInt = UInt(this)

/**
 * Converts this [Long] value to [UInt].
 *
 * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Long` value.
 * The resulting `UInt` value is represented by the least significant 32 bits of this `Long` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toUInt(): UInt = UInt(this.toInt())

/**
 * Converts this [Float] value to [UInt].
 *
 * The fractional part, if any, is rounded down towards zero. Returns zero if this `Float` value is negative or `NaN`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toUInt(): UInt = doubleToUInt(this.toDouble())

/**
 * Converts this [Double] value to [UInt].
 *
 * The fractional part, if any, is rounded down towards zero. Returns zero if this `Double` value is negative or `NaN`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toUInt(): UInt = doubleToUInt(this)
```

```
// Auto-generated file. DO NOT EDIT!

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Open source used in DNAC Platform Arica

// Since Kotlin: "1.3"
// ExperimentalUnsignedTypes

// Kotlin

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UShort @PublishedApi internal constructor(@PublishedApi internal val data: Short) : Comparable<UShort> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UShort can have.
         */
        public const val MIN_VALUE: UShort = UShort(0)

        /**
         * A constant holding the maximum value an instance of UShort can have.
         */
        public const val MAX_VALUE: UShort = UShort(-1)

        /**
         * The number of bytes used to represent an instance of UShort in a binary form.
         */
        public const val SIZE_BYTES: Int = 2

        /**
         * The number of bits used to represent an instance of UShort in a binary form.
         */
        public const val SIZE_BITS: Int = 16
    }

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UShort): Int = this.toInt().compareTo(other.toInt())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
     */
    @kotlin.internal.InlineOnly
```
public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)/**
 * Compares this value with the specified value for order.
 * * Returns zero if this value is equal to the specified other value, a
 * negative number if it's less than other,
 * * or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): Int =
this.toULong().compareTo(other)/** Adds the other value to this value.*/
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())/** Adds the other value to this value.*/
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UShort): UInt =
this.toUInt().plus(other.toUInt())/** Adds the other value to this value.*/
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UInt): UInt =
this.toUInt().plus(other)/** Adds the other value to this value.*/
@kotlin.internal.InlineOnly
public inline operator fun plus(other: ULong): ULong =
this.toULong().plus(other)/** Adds the other value to this value.*/
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())/** Subtracts the other value from this value.*/
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UShort): UInt =
this.toUInt().minus(other.toUInt())/** Subtracts the other value from this value.*/
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): UInt =
this.toUInt().minus(other)/** Subtracts the other value from this value.*/
@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong =
this.toULong().minus(other)/** Subtracts the other value from this value.*/
@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): UInt =
this.toUInt().times(other.toUInt())/** Multiplies this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): UInt =
this.toUInt().times(other.toUInt())/** Multiplies this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): UInt =
this.toUInt().times(other)/** Multiplies this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong =
this.toULong().times(other)/** Multiplies this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): UInt =
this.toUInt().div(other.toUInt())/** Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): UInt =
this.toUInt().div(other.toUInt())/** Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): UInt =
this.toUInt().div(other)/** Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong =
this.toULong().div(other)/** Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): UInt =
this.toUInt().rem(other.toUInt())/** Calculates the remainder of
* * Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt =
this.toUInt().rem(other.toUInt())/** Calculates the remainder of
* * Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt =
this.toUInt().rem(other)/** Calculates the remainder of
* * Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong =
this.toULong().rem(other)/** Calculates the remainder of
* * Divides this value by the other value.*/
@kotlin.internal.InlineOnly
public inline operator fun inc(): UShort = UShort(data.inc())/** Creates a range from this value to the specified [other] value.*/
@kotlin.internal.InlineOnly
public inline operator fun dec(): UShort = UShort(data.dec())/** Creates a range from this value to the specified [other] value.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
* * Calculates the remainder of the two values.*/
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange =
UIntRange(this.toUInt(), other.toUInt())/** Performs a bitwise
than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents
* the same numerical value as this `UShort`.\n *\n * The resulting `Byte` value is represented by the least significant 8 bits of this `UShort` value.\n * Note that the resulting `Byte` value may be negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toByte(): Byte = data.toByte()\n /**\n * Converts this [UShort] value to [Byte].\n */\n * If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents\n * the same numerical value as this `UShort`. Otherwise the result is negative.\n */\n @kotlin.internal.InlineOnly\n public inline fun toShort(): Short = data\n /**\n * Converts this [UShort] value to [Int].\n */\n * The resulting `Int` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `Int` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toInt(): Int = data.toInt() and 0xFFFF\n /**\n * Converts this [UShort] value to [Long].\n */\n * The resulting `Long` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `Long` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toLong(): Long = data.toLong() and 0xFFFF\n /**\n * Converts this [UShort] value to [UByte].\n */\n * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents\n * the same numerical value as this `UShort`.\n *\n * The resulting `UByte` value is represented by the least significant 8 bits of this `UShort` value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUByte(): UByte = data.toUByte()\n /**\n * Returns this value.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUShort(): UShort = data\n /**\n * Converts this [UShort] value to [UInt].\n */\n * The resulting `UInt` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toUInt(): UInt = data.toInt() and 0xFFFF\n /**\n * Converts this [UShort] value to [ULong].\n */\n * The resulting `ULong` value represents the same numerical value as this `UShort`.\n *\n * The least significant 16 bits of the resulting `ULong` value are the same as the bits of this `UShort` value,\n * whereas the most significant 16 bits are filled with zeros.\n */\n @kotlin.internal.InlineOnly\n public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF)\n /**\n * Converts this [UShort] value to [Double].\n */\n * The resulting `Double` value represents the same numerical value as this `UShort`.\n */\n @kotlin.internal.InlineOnly\n public inline fun toFloat(): Float = this.toFloat()\n /**\n * Converts this [UShort] value to [Float].\n */\n @kotlin.internal.InlineOnly\n public override fun toString(): String = toInt().toString()
16 bits of this 'Long' value.

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\public inline fun Long.toUShort(): UShort = UShort(this.toShort())\n","/**\n * A range of values of type `Char`.\n */\npublic class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {\n override val start: Char get() = first\n override val endInclusive: Char get() = last\n override fun contains(value: Char): Boolean = first <= value && value <= last\n override fun isEmpty(): Boolean = first > last\n override fun equals(other: Any?): Boolean =\n other is CharRange && (isEmpty() || first == other.first && last == other.last)\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * first.toInt() + last.toInt())\n \n /**\n * A range of values of type `Int`.\n */\npublic class IntRange(start: Int, endInclusive: Int) : IntProgression(start, endInclusive, 1), ClosedRange<Int> {\n override val start: Int get() = first\n override val endInclusive: Int get() = last\n override fun contains(value: Int): Boolean = first <= value && value <= last\n override fun isEmpty(): Boolean = first > last\n override fun equals(other: Any?): Boolean =\n other isEmpty() && other.equals(first, last)\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * first + last)\n \n /**\n * A range of values of type `Long`.\n */\npublic class LongRange(start: Long, endInclusive: Long) : LongProgression(start, endInclusive, 1), ClosedRange<Long> {\n override val start: Long get() = first\n override val endInclusive: Long get() = last\n override fun contains(value: Long): Boolean = first <= value && value <= last\n override fun isEmpty(): Boolean = first > last\n override fun equals(other: Any?): Boolean =\n other isEmpty() && other.equals(first, last)\n override fun hashCode(): Int =\n if (isEmpty()) -1 else (31 * first xor (first ushr 32)) + (last xor (last ushr 32))\n override fun toString(): String = "$first..$last"
T>.asCollection(): Collection<T> = ArrayAsCollection(this, isVarargs = false)

private class
    ArrayAsCollection<T>(val values: Array<out T>, val isVarargs: Boolean) : Collection<T> {
        override val size: Int get() = values.size
        override fun isEmpty(): Boolean = values.isEmpty()
        override fun contains(element: T): Boolean = values.contains(element)
        override fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) }
        override fun iterator(): Iterator<T> = values.iterator()
        // override hidden toArray implementation to prevent copying of values array
        override fun toArray(): Array<out Any?> = values.copyToArrayOfAny(isVarargs)
    }

/**
 * Returns an empty read-only list. The returned list is serializable (JVM).
 */
public fun <T> emptyList(): List<T> = EmptyList

/**
 * Returns a new read-only list with the given elements.
 */
public fun <T> listOf(vararg elements: T): List<T> = if (elements.size > 0) elements.asList() else emptyList()

/**
 * Returns an empty new [MutableList].
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> mutableListOf(): MutableList<T> = ArrayList()

/**
 * Returns a new [ArrayList] with the specified [size], where each element is calculated by calling the specified
 * [init] function.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> MutableList(size: Int, init: (index: Int) -> T): MutableList<T> =
    if (size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs = true))

/**
 * Returns an index of the last item in the list or -1 if the list is empty.
 */
public val <T> List<T>.lastIndex: Int
    get() = this.size - 1

/**
 * Returns true if the collection is not empty.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.isEmpty(): Boolean = !isEmpty()

/**
 * Returns true if this nullable collection is
either null or empty.

@sample samples.collections.Collections.Collections.collectionIsNull

* @kotlin.internal.InlineOnly

public inline fun <T> Collection<T>?.isNull(): Boolean {
    contract {
        returns(false) implies (this@isNull != null)
    }

    return this == null ||
    this.isEmpty()
}

/**
 * Returns this Collection if it's not `null` and the empty list otherwise.
 * @sample samples.collections.Collections.Collections.collectionIsNullOrEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isEmpty(): Boolean {
    contract {
        returns(false) implies (this@isEmpty != null)
    }

    return this == null ||
    this.isEmpty()
}

/**
 * Returns this Collection if it's not `null` and the empty list otherwise.
 * @sample samples.collections.Collections.Collections.collectionIsNullOrEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isEmpty(): Boolean {
    contract {
        returns(false) implies (this@isEmpty != null)
    }

    return this == null ||
    this.isEmpty()
}

/**
 * Returns this Collection if it's not `null` and the empty list otherwise.
 * @sample samples.collections.Collections.Collections.collectionIsNullOrEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isEmpty(): Boolean {
    contract {
        returns(false) implies (this@isEmpty != null)
    }

    return this == null ||
    this.isEmpty()
}

/**
 * Checks if all elements in the specified collection are contained in this collection.
 * * Allows to overcome type-safety restriction of `containsAll` that requires to pass a collection of type `Collection<E>`.
 * @sample samples.collections.Collections.Collections.collectionIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Collection<*>, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Searches this list or its range for the provided [element] using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the Composable natural ordering of its elements.
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.
 * * `null` value is considered to be less than any non-null value.
 * * @return the index of the element, if it is contained in the list within the specified range;
 * * otherwise, the inverted insertion point `(insertion point - 1)`. The insertion point is defined as the index at which the element should be inserted.
 * * so that the list (or the specified subrange of list) still remains sorted.
 * * @sample samples.collections.Collections.Collections.collectionContainsAll
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)

internal fun <T> List<T>.optimizeReadOnlyList() = when (size) {
    0 -> emptyList()
    1 -> listOf(this[0])
    else ->
        this
}

/**
 * Searches this list or its range for an element having the key returned by the specified [selector] function
 * equal to the provided [key] value using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the specified [comparator].
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.
 * * `null` value is considered to be less than any non-null value.
 * * @return the index of the element, if it is contained in the list within the specified range;
 * * otherwise, the inverted insertion point `(insertion point - 1)`. The insertion point is defined as the index at which the element should be inserted.
 * * so that the list (or the specified subrange of list) still remains sorted.
 * * @sample samples.collections.Collections.Collections.collectionContainsAll
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
public inline fun <T> List<T>.binarySearch(element: T, comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparator.compare(midVal, element)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
listing page
EmptyList
private fun readResolve(): Any = EmptyMap

/**
 * Returns an empty read-only map of specified type.
 * The returned map is serializable (JVM).
 */
@kotlin.internal.InlineOnly
public fun <K, V> emptyMap(): Map<K, V> = EmptyMap<br/>

/**
 * Returns a new read-only map with the specified contents, given as a list of pairs
 * where the first value is the key and the second is the value.
 * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
 */
@kotlin.internal.InlineOnly
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =
    pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))

/**
 * Returns a new read-only map.
 * The returned map is serializable (JVM).
 */
@kotlin.internal.InlineOnly
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> =
    pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))

/**
 * Calculate the initial capacity of a map, based on Guava's com.google.common.collect.Maps approach. This is
 * equivalent to the Collection constructor for HashSet, (c.size()/0.75f) + 1, but provides further optimisations for
 * very small or very large sizes, allows support non-collection classes, and provides consistency for all map based
 * class construction.
 */
@SinceKotlin("1.1")
@PublishedApi
internal fun mapCapacity(expectedSize: Int): Int {
    if (expectedSize < 3) {
        return expectedSize + 1
    }
    if (expectedSize < INT_MAX_POWER_OF_TWO) {
        return expectedSize + expectedSize / 3
    }
    return Int.MAX_VALUE // any large value
}

private const val INT_MAX_POWER_OF_TWO: Int = Int.MAX_VALUE / 2 + 1
Open Source Used In DNAC Platform Arica 12079

```kotlin
return this == null || isEmpty()
}

/**
 * Returns the `Map` if its not `null`, or the empty `Map` otherwise.
 * Returns the this map if it's not empty or the result of calling `defaultValue` function if the map is empty.
 * @param key The key of the entry.
 * @param defaultValue The value to return if the key is not present.
 * @return The value corresponding to the given key, or `null` if such a key is not present in the map.
 */
@kotlin.internal.InlineOnly
public <K, V> Map<K, V> getOrElse(K key, V defaultValue)
{
    return this.get(key) ?: defaultValue()
}

/**
 * Converts entry to `Pair` with key being first component and value being second.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map.Entry<K, V>.toPair(): Pair<K, V> = Pair(this.component1(), this.component2())

/**
 * Returns the value for the given key, or throws an exception if there was no entry for the given key.
 * @param key The key of the entry.
 * @return The value for the given key.
 */
@kotlin.internal.InlineOnly
{
    return this.getOrElse(key, defaultValue)
}

/**
 * Converts entry to [Pair] with key being first component and value being second.
 * Returns the value for the given key, or throws an exception if there was no entry for the given key.
 * @param key The key of the entry.
 * @param defaultValue The value to return if the key is not present.
 * @return The value for the given key.
 */
@kotlin.internal.InlineOnly
{
    return getOrImplicitDefault(key, defaultValue)
}
```

Open Source Used In DNAC Platform Arica 12079
the [defaultValue] function,\n * puts its result into the map under the given key and returns it.\n * Note that the operation is not guaranteed to be atomic if the map is being modified concurrently.\n * @sample \n * @sample samples.collections.Maps.Usage.getOrPut
 */
public inline fun <K, V> MutableMap<K, V>.getOrPut(key: K,\n defaultValue: () -> V): V {\n    val value = get(key)\n    return if (value == null) {\n        val answer = defaultValue()\n        put(key, answer)\n        answer\n    } else {\n        value\n    }\n}

/**
 * Returns an [Iterator] over the entries in the [Map].\n */
@kotlin.internal.InlineOnly

/**
 * Returns a [MutableIterator] over the mutable entries in the [MutableMap].\n */
@kotlin.jvm.JvmName("mutableIterator")
@kotlin.internal.InlineOnly

/**
 * Populates the given [destination] map with entries having the keys of this map and the values obtained by applying the [transform] function to each entry in this [Map].\n */
public inline fun <K, V, R, M : MutableMap<in R, in V>> Map<out K, V>.mapValuesTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {\n    return entries.associateByTo(destination, transform, { it.value })\n}

/**
 * Populates the given [destination] map with entries having the keys obtained by applying the [transform] function to each entry in this [Map],\n * and the values of this map.\n */
public inline fun <K, V, R, M : MutableMap<in R, in V>> Map<out K, V>.mapKeysTo(destination: M, transform: (Map.Entry<K, V>) -> R): M {\n    return entries.associateByTo(destination, transform, { it.value })\n}

/**
 * Returns a new map with entries having the keys of this map and the values obtained by applying the [transform] function to each entry in this [Map].\n */
public inline fun <K, V, R> Map<out K, V>.mapValues(transform: (Map.Entry<K, V>) -> R): Map<K, R> {\n    return mapValuesTo(LinkedHashMap<K, R>(mapCapacity(size)), transform)\n}

/**
 * Returns a new Map with entries having the keys obtained by applying the [transform] function to each entry in this [Map],\n * and the values of this map.\n */
public inline fun <K, V, R> Map<out K, V>.mapKeys(transform: (Map.Entry<K, V>) -> R): Map<R, V> {\n    return mapKeysTo(LinkedHashMap<R, V>(mapCapacity(size)), transform)\n}

/**
 * Returns a map containing all key-value pairs with keys matching the given [predicate].\n */
public inline fun <K, V> Map<out K, V>.filterKeys(predicate: (K) -> Boolean): Map<K, V> {\n    val result = LinkedHashSet<Map.Entry<K, V>>(\n        for (entry in this) {\n            if (predicate(entry.key)) {\n                result.put(entry.key, entry.value)\n            }\n        }\n    )\n    return result\n}

/**
 * Returns a map containing all key-value pairs with values matching the given [predicate].\n */
public inline fun <K, V> Map<out K, V>.filterValues(predicate: (V) -> Boolean): Map<K, V> {\n    val result = LinkedHashSet<Map.Entry<K, V>>(\n        for (entry in this) {\n            if (predicate(entry.value)) {\n                result.put(entry.key, entry.value)\n            }\n        }\n    )\n    return result\n}
parameter.

*public inline fun <K, V, M : MutableMap<K, V>, in V> filterTo(destination: M, predicate: (Map.Entry<K, V> -> Boolean)): M { if (predicate(element)) { destination.put(element.key, element.value) return destination } else return destination } * Returns a new map containing all key-value pairs matching the given [predicate].

*public inline fun <K, V> filterTo(destination: M, predicate: (Map.Entry<K, V> -> Boolean)): Map<K, V> { if (predicate(destination)) return destination } * Appends all entries not matching the given [predicate] into the given [destination].

*public inline fun <K, V, M : MutableMap<K, V>, in V> filterTo(destination: M, predicate: (Map.Entry<K, V> -> Boolean)): M { if (predicate(destination)) { destination.put(destination.key, destination.value) return destination } else return destination } * Returns a new map containing all key-value pairs not matching the given [predicate].

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Map<out K, V>.filter(predicate: (Map.Entry<K, V> -> Boolean)): Map<K, V> { return filterTo(LinkedHashMap<K, V>(), predicate) } * Returns a new map containing all key-value pairs from the given collection of pairs.

* The returned map preserves the entry iteration order of the original collection. If any of two pairs would have the same key the last one gets added to the map.

*public inline fun <K, V> Iterable<Pair<K, V>>.toMap(): Map<K, V> { return toMap(LinkedHashMap<K, V>(), mapCapacity(size)) } * Populates and returns the [destination] mutable map with key-value pairs from the given collection of pairs.

* The returned map preserves the entry iteration order of the original array. If any of two pairs would have the same key the last one gets added to the map.

*public inline fun <K, V> Array<Pair<K, V>>.toMap(): Map<K, V> = when (size) { 0 -> emptyMap() 1 -> mapOf(this[0]) else -> toMap(LinkedHashMap<K, V>(mapCapacity(size))) } * Returns a new map containing all key-value pairs from the given collection of pairs.

* The returned map preserves the entry iteration order of the original array.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap() * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V, M : MutableMap<K, V>, in V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> mapOf(this[0]) else -> toMap(LinkedHashMap<K, V>(mapCapacity(size))) } * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V, M : MutableMap<K, V>, in V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new map containing all key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V, M : MutableMap<K, V>, in V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()) * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new map containing all key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()) * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()) * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()) * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>()) * Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

* The returned map preserves the entry iteration order of the original sequence.

*public inline fun <K, V> Sequence<Pair<K, V>>.toMap(destination: M): M =when (size) { 0 -> emptyMap() 1 -> toSingletonMap() else -> toMutableMap() } * Returns a new read-only map containing all key-value pairs from the original map.

* The returned map preserves the entry iteration order of the original map.
Pair<K, V>): Map<K, V> =
    if (this.isEmpty()) mapOf(pair) else LinkedHashMap(this).apply { put(pair.first, pair.second) }

/**
 * Creates a new read-only map by replacing or adding entries to this map from a given array of key-value [pairs].
 */
    if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }

/**
 * Creates a new read-only map by replacing or adding entries to this map from a given sequence of key-value [pairs].
 */
    if (this.isEmpty()) pairs.toMap() else LinkedHashMap(this).apply { putAll(pairs) }

/**
 * Creates a new read-only map by replacing or adding entries to this map from another [map].
 */
    LinkedHashMap(this).apply { putAll(map) }

/**
 * Appends or replaces the given [pair] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pair: Pair<K, V>) {
    put(pair.first, pair.second)
}

/**
 * Appends or replaces all pairs from the given collection of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Appends or replaces all pairs from the given array of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Array<out Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Appends or replaces all pairs from the given sequence of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Sequence<Pair<K, V>>) {
    putAll(pairs)
}

/**
 * Removes the entry with the given [key] from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(key: K) {
    remove(key)
}

/**
 * Removes all entries the keys of which are contained in the given [keys] collection from this mutable map.
 */
@SinceKotlin("1.1")
public inline operator fun <K, V>MutableMap<in K, in V>.minus(keys: Collection<K>) {
    removeAll(keys)
}.optimizeReadOnlyMap()
package kotlin.collections

internal object EmptySet : Set<Nothing>, Serializable {
    private const val serialVersionUID: Long = 3406603774387020532

    override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty()
    override fun hashCode(): Int = 0
    override fun toString(): String = "[]"

    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun contains(element: Nothing): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()

    override fun iterator(): Iterator<Nothing> = EmptyIterator

    private fun readResolve(): Any = EmptySet
}

/**
 * Returns an empty read-only set. The returned set is serializable.
 * @sample samples.collections.Collections.Sets.emptyReadOnlySet
 */
public fun <T> emptySet(): Set<T> = EmptySet

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable.
 * @sample samples.collections.Collections.Sets.readOnlySet
 */
public fun <T> setOf(vararg elements: T): Set<T> = if (elements.size > 0) elements.toSet() else emptySet()

/**
 * Returns an empty read-only set. The returned set preserves the element iteration order.
 * @sample samples.collections.Collections.Sets.emptyList
 */
public fun <T> mutableSetOf(): MutableSet<T> = LinkedHashSet()

/**
 * Returns a new read-only set with the given elements.
 * @sample samples.collections.Collections.Sets.mutableSet
 */
public fun <T> hashSetOf(vararg elements: T): HashSet<T> = elements.toCollection(HashSet(mapCapacity(elements.size)))

/** Returns an empty new [LinkedHashSet].
 * @sample samples.collections.Collections.Sets.hashSet
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> linkedSetOf(): LinkedHashSet<T> = LinkedHashSet()

/** Returns this Set if it's not `null` and the empty set otherwise.
 * @sample samples.collections.Collections.Sets.orEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>?.orEmpty(): Set<T> = this ?: emptySet()

internal fun <T> Collections.Set<T>.optimizeReadOnlySet() = when (size) {
    0 -> emptySet()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Set<T>.optimizeMutableSet() = when (size) {
    0 -> emptySet()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Set<T>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Map<T, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Map<T, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

@PublishedApi
internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("\SetsKt\")

package kotlin.collections

internal object EmptySet : Set<Nothing>, Serializable {
    private const val serialVersionUID: Long = 3406603774387020532

    override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty()
    override fun hashCode(): Int = 0
    override fun toString(): String = "[]"

    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun contains(element: Nothing): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()

    override fun iterator(): Iterator<Nothing> = EmptyIterator

    private fun readResolve(): Any = EmptySet
}

/**
 * Returns an empty read-only set. The returned set is serializable.
 * @sample samples.collections.Collections.Sets.emptyReadOnlySet
 */
public fun <T> emptySet(): Set<T> = EmptySet

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable.
 * @sample samples.collections.Collections.Sets.readOnlySet
 */
public fun <T> setOf(vararg elements: T): Set<T> = if (elements.size > 0) elements.toSet() else emptySet()

/**
 * Returns an empty read-only set. The returned set preserves the element iteration order.
 * @sample samples.collections.Collections.Sets.emptyList
 */
public fun <T> mutableSetOf(): MutableSet<T> = LinkedHashSet()

/**
 * Returns a new read-only set with the given elements.
 * @sample samples.collections.Collections.Sets.mutableSet
 */
public fun <T> hashSetOf(vararg elements: T): HashSet<T> = elements.toCollection(HashSet(mapCapacity(elements.size)))

/** Returns an empty new [LinkedHashSet].
 * @sample samples.collections.Collections.Sets.hashSet
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> linkedSetOf(): LinkedHashSet<T> = LinkedHashSet()

/** Returns this Set if it's not `null` and the empty set otherwise.
 * @sample samples.collections.Collections.Sets.orEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>?.orEmpty(): Set<T> = this ?: emptySet()

internal fun <T> Collections.Set<T>.optimizeReadOnlySet() = when (size) {
    0 -> emptySet()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Set<T>.optimizeMutableSet() = when (size) {
    0 -> emptySet()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Set<T>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

internal fun <T> Collections.Map<T, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

@PublishedApi
internal fun <K, V> Map<K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("\StringsKt\")

package kotlin.text

/**
 * Parses the string as a signed [Byte]
Open Source Used In DNAC Platform Arica 12084

Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that
can be found in the license/LICENSE.txt file.

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.

@file:kotlin.jvm.JvmName("CollectionsKt")

*/
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

/**
 * Given an [iterator] function constructs an [Iterable] instance that returns values through
 * the [Iterator] provided by that function.
 * @sample samples.collections.Iterables.Building.iterable
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
    override fun iterator(): Iterator<T> = iterator()
}

/**
 * A wrapper over another [Iterable] (or any other object that can produce an [Iterator]) that returns
 * an indexing iterator.
 */
internal class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
    override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())
}

/**
 * Returns the size of this iterable if it is known, or `null` otherwise.
 */
@PublishedApi
internal fun <T>
Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }

* * Returns a sequence of all elements from each iterable in this sequence.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* The operation is _intermediate_ and _stateless_.

* The `flatten` function is used to flatten nested sequences.

```kotlin
fun Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }
```

Sequence<Iterable<T>>.flatten(): Sequence<T>

* The operation is _terminal_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun Sequence<Iterable<T>>.flatten(): Sequence<T> = flatten { it.iterator() }
```

Sequence<R> { in if (this is TransformingSequence<*<*>> { return (this as TransformingSequence<*<*>>)

* The operation is _intermediate_ and _stateless_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun Sequence<R> { in if (this is TransformingSequence<*<*>> { return (this as TransformingSequence<*<*>>)
```

Sequence<Pair<T, R>>.unzip(): Sequence<Pair<T, R>>

* The operation is _terminal_.

* An overridden function that takes an input of type Sequence<Pair<T, R>> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun Sequence<Pair<T, R>>.unzip(): Sequence<Pair<T, R>>
```

FilteringSequence<T, R>.

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun FilteringSequence<T, R>
```

TransformingIndexedSequence<T, R>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun TransformingIndexedSequence<T, R>
```

TransformingSequence<T, R>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun TransformingSequence<T, R>
```

FilteringSequence<T, R>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun FilteringSequence<T, R>
```

IndexedValue<T>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun IndexedValue<T>
```

TransformingIndexedValue<T>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun TransformingIndexedValue<T>
```

TransformingValue<T>

* The operation is _intermediate_.

* An overridden function that takes an input of type Sequence<T> and returns a new sequence that is the flattened version of the input sequence.

* Values of each pair from this sequence.

```kotlin
fun TransformingValue<T>
```
override fun hasNext(): Boolean {
    if (nextState == -1)
        calcNext() // will change nextState
    return nextState == 1
}

override fun drop(n: Int): Sequence<T> = (count + n).let {
    n1 -> if (n1 < 0) DropSequence(this, n) else DropSequence(sequence, n1)
}

override fun take(n: Int): Sequence<T> = (count + n).let {
    n1 -> if (n1 < 0) TakeSequence(this, n) else SubSequence(sequence, count, n1)
}

override fun iterator(): Iterator<T> = object : Iterator<T> {
    val iterator = sequence.iterator()
    var left = count

    private fun drop() {
        while (left > 0 && iterator.hasNext()) {
            iterator.next()
            left--
        }
    }

    override fun next(): T {
        drop()
        return iterator.next()
    }

    override fun hasNext(): Boolean {
        drop()
        return iterator.hasNext()
    }
}

override fun hasNext(): Boolean {
    if (nextState == -1)
        calcNext() // will change nextState
    return nextState == 1
}

@Suppress("UNCHECKED_CAST")
val result = nextItem as T

return result

override fun computeNext() {
    while (source.hasNext()) {
        val next = source.next()
        val key = keySelector(next)

        if (observed.add(key)) {
            setNext(next)
            return
        }
    }

    done()
}

override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
}

override fun override fun hasNext(): Boolean {
    return hasNext()
fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {
    return GeneratorSequence(nextFunction, { nextFunction() }).constrainOnce()
}

/**
 * Returns a sequence defined by the function [seedFunction], which
 * is invoked to produce the starting value, and the [nextFunction], which
 * is invoked to calculate the next value based on the previous one on each iteration.
 * The sequence produces values until it encounters first `null` value.
 * If [seed] is `null`, an empty sequence is produced.
 * The sequence can be iterated multiple times, each time starting with [seed].
 */
@kotlin.internal.LowPriorityInOverloadResolution
public fun <T : Any> generateSequence(seed: T?, nextFunction: (T) -> T?): Sequence<T> =
    if (seed == null) EmptySequence
    else GeneratorSequence({ seed }, nextFunction)

/**
 * Returns a sequence defined by the function [seedFunction], which
 * is invoked to produce the starting value,
 * and the [nextFunction], which is invoked to calculate the next
 * value based on the previous one on each iteration.
 * The sequence produces values until it encounters first `null` value.
 * If [seedFunction] returns `null`, an empty sequence is produced.
 * The sequence can be iterated multiple times.
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> require(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    require(value) { "Failed requirement." }
}

/**
 * Throws an [IllegalArgumentException] if the [value] is false.
 * @sample samples.misc.Preconditions.failRequireWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun require(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    }
}

/**
 * Returns the not null value if the [value] is null.
 * Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun check(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    check(value) { "Check failed." }
}

/**
 * Throws an [IllegalArgumentException] if the [value] is false.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun check(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    }
}

/**
 * Returns the not null value if the [value] is null.
 * Otherwise returns the not null value.
 */
@kotlin.internal.InlineOnly
public inline fun <T : Any> requireNotNull(value: T?): T {
    contract {
        returns() implies (value != null)
    }
    return requireNotNull(value) { "Required value was null." }
}

/**
 * Throws an [IllegalArgumentException] if the [value] is null.
 * Otherwise returns the not null value.
 */
@kotlin.internal.InlineOnly
    contract {
        returns() implies (value != null)
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] if the [value] is false.
 */
@kotlin.internal.InlineOnly
public inline fun checkNotNull(value: T?): T {
    contract {
        returns() implies value != null
    }
    return checkNotNull(value) { "Required value was null." }
}

/**
 * Throws an [IllegalStateException] if the [value] is false.
 */
@kotlin.internal.InlineOnly
public inline fun checkNotNull(value: T?, lazyMessage: () -> Any): T {
    contract {
        returns() implies value != null
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] if the [value] is null.
 */
@kotlin.internal.InlineOnly
public inline fun checkNotNull(value: T?, lazyMessage: () -> Any): T {
    contract {
        returns() implies value != null
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    }
}
samples.misce.Preconditions.failCheckWithLazyMessage

//
// NOTE: THIS FILE IS AUTO-GENERATED
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*
import primitiveArrayConcat
import withType
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun <T> Array<out T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun ByteArray.elementAt(index: Int): Byte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun ShortArray.elementAt(index: Int): Short {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun IntArray.elementAt(index: Int): Int {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun LongArray.elementAt(index: Int): Long {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun FloatArray.elementAt(index: Int): Float {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun DoubleArray.elementAt(index: Int): Double {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */
public actual fun BooleanArray.elementAt(index: Int): Boolean {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.
 */
public actual fun <T> Array<out T>.asList(): List<T> {
    return ArrayList<T>(this.unsafeCast<Array<Any?>>())
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun ByteArray.asList(): List<Byte> {
    return this.unsafeCast<Array<Byte>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun ShortArray.asList(): List<Short> {
    return
this.unsafeCast<Array<Short>>().asList().

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun IntArray.asList(): List<Int> {
    return this.unsafeCast<Array<Int>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun LongArray.asList(): List<Long> {
    return this.unsafeCast<Array<Long>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun FloatArray.asList(): List<Float> {
    return this.unsafeCast<Array<Float>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun DoubleArray.asList(): List<Double> {
    return this.unsafeCast<Array<Double>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
@kotlin.internal.InlineOnly
public actual inline fun BooleanArray.asList(): List<Boolean> {
    return this.unsafeCast<Array<Boolean>>().asList()
}

/**
 * Returns a [List] that wraps the original array.
 */
public actual fun CharArray.asList(): List<Char> {
    return object : AbstractList<Char>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: Char): Boolean = this@asList.contains(element)
        override fun get(index: Int): Char {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: Char): Int {
            if ((element as Any?) !is Char) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: Char): Int {
            if ((element as Any?) !is Char) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns `true` if the two specified arrays are *deeply* equal to one another, \n * i.e. contain the same number of the same elements in the same order. \n * * * If two corresponding elements are nested arrays, they are also compared deeply. \n * * If any of arrays contains itself on any nesting level the behavior is undefined. \n * * The elements of other types are compared for equality with the [equals][Any.equals] function. \n * * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `-0.0`. \n */
@SinceKotlin("1.1")
@library("arrayDeepEquals")
public actual infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean {
    definedExternally
}

/**
 * Returns a hash code based on the contents of this array as if it is [List]. \n * Nested arrays are treated as lists too. \n * * If any of arrays contains itself on any nesting level the behavior is undefined. \n */
@SinceKotlin("1.1")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    definedExternally
}

/**
 * Returns a string representation of the contents of this array as if it is a [List]. \n * Nested arrays are treated as lists too. \n * * If any of arrays contains itself on any nesting level the behavior is undefined. \n */
@SinceKotlin("1.1")
@library("arrayDeepToString")
public actual fun <T> Array<out T>.contentDeepToString(): String {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, \n * i.e. contain the same number of the same elements in the same order. \n * * * The elements are compared for equality with the [equals][Any.equals] function. \n * * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `-0.0`. \n */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, \n * i.e. contain the same number of the same elements in the same order. \n * * * The elements are compared for equality with the [equals][Any.equals] function. \n * * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `-0.0`. \n */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, \n * i.e. contain the same number of the same elements in the same order. \n * * * The elements are compared for equality with the [equals][Any.equals] function. \n * * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `-0.0`. \n */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun ShortArray.contentEquals(other: ShortArray): Boolean {
    definedExternally
}
ShortArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are
 * structurally equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun IntArray.contentEquals(other: IntArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun LongArray.contentEquals(other: LongArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are
 * structurally equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun FloatArray.contentEquals(other: FloatArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are
 * structurally equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are
 * structurally equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are
 * structurally equal to one another, i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun CharArray.contentEquals(other: CharArray): Boolean {
    definedExternally
}
CharArray.contentHashCode(): Int
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun <T> Array<out T>.contentToString():
String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun ByteArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun ShortArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun IntArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun LongArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun FloatArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun DoubleArray.contentToString(): String
        definedExternally
        * Returns a string representation of the contents of the specified array as if it is [List]<\n        * @sample samples.collections.Arrays.ContentOperations

*\n@SinceKotlin("1.1")\n@unlibrary("arrayToString")
unpublic actual fun BooleanArray.contentToString(): String
        definedExternally
        * Copies this array or its subrange into the [destination] array and returns that array.\n        * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n        * @param destination the array to copy to.\n        * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n        * @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.\n        * @param endIndexOf the end (exclusive) of the subrange to copy, size of this array by default.\n        * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndexOf] or [endIndexOf] is out of range of this array indices or when \`startIndexOf > endIndexOf\`\n        * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n        * or when that index is out of the [destination] array indices range.\n        * @return the [destination] array.\n
*\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress(\"ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS\")
unpublic actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndexOf: Int = 0, endIndexOf: Int = size): Array<T>
        copy into the destination. Array<T>, destinationOffset: Int = 0, startIndexOf: Int = 0, endIndexOf: Int = size: Array<T>\n        * Copies this array or its subrange into the [destination] array and returns that array.\n        * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.\n        * @param destination the array to copy to.\n        * @param destinationOffset the position in the [destination] array to copy to, 0 by default.\n        * @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.\n        * @param endIndexOf the end (exclusive) of the subrange to copy, size of this array by default.\n        * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndexOf] or [endIndexOf] is out of range of this array indices or when \`startIndexOf > endIndexOf\`\n        * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].\n        * or when that index is out of the [destination] array indices range.\n        * @return the [destination] array.
> endIndex.\n * @throws IndexOutOfBoundsException when the subrange doesn’t fit into the [destination] array starting at the specified [destinationOffset],\n * or when that index is out of the [destination] array indices range.\n * @return the [destination] array.\n *\n * @SinceKotlin("1.3")\n * @kotlin.internal.InlineOnly\n * @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, 
startIndex: Int = 0, endIndex: Int = size): ByteArray {\n    arrayCopy(this.unsafeCast<Array<Byte>>(), 
destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)\n    return destination
}\n
*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, 
startIndex: Int = 0, endIndex: Int = size): ShortArray {\n    arrayCopy(this.unsafeCast<Array<Short>>(), 
destination.unsafeCast<Array<Short>>(), destinationOffset, startIndex, endIndex)\n    return destination
}\n
*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, 
startIndex: Int = 0, endIndex: Int = size): IntArray {\n    arrayCopy(this.unsafeCast<Array<Int>>(), 
destination.unsafeCast<Array<Int>>(), destinationOffset, startIndex, endIndex)\n    return destination
}\n
*/
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, 
startIndex: Int = 0, endIndex: Int = size): LongArray {\n    arrayCopy(this.unsafeCast<Array<Long>>(), 
destination.unsafeCast<Array<Long>>(), destinationOffset, startIndex, endIndex)\n    return destination
}\n
*/
@param destination the array to copy to.
@param destinationOffset the position in the [destination] array to copy to, 0 by default.
@param startIdx the beginning (inclusive) of the subrange to copy, size of this array by default.
@param endIdx the end (exclusive) of the subrange to copy, size of this array by default.

Throws IndexOutOfBoundsException or IllegalArgumentException when [startIdx] or [endIdx] is out of range of this array indices or when `startIdx > endIdx`.

@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
public actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndest: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndest, endIndex)
    return destination
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual inline fun CharArray.copyOf(): CharArray {
    return this.asDynamic().slice().unsafeCast<CharArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun ByteArray.copyOf(): ByteArray {
    return this.asDynamic().slice().unsafeCast<ByteArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun ShortArray.copyOf(): ShortArray {
    return this.asDynamic().slice().unsafeCast<ShortArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun IntArray.copyOf(): IntArray {
    return this.asDynamic().slice().unsafeCast<IntArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun LongArray.copyOf(): LongArray {
    return this.asDynamic().slice().unsafeCast<LongArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun FloatArray.copyOf(): FloatArray {
    return this.asDynamic().slice().unsafeCast<FloatArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun DoubleArray.copyOf(): DoubleArray {
    return this.asDynamic().slice().unsafeCast<DoubleArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun BooleanArray.copyOf(): BooleanArray {
    return this.asDynamic().slice().unsafeCast<BooleanArray>()
}

/**
* Returns new array which is a copy of the original array.
* @sample samples.collections.Arrays.CopyOfOperations.copyOf
*/
@Suppress("NOTHING_TO_INLINE")
public actual inline fun CharArray.copyOf(): CharArray {
    return this.asDynamic().slice().unsafeCast<CharArray>()
}
resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*anonymous actual fun LongArray.copyOf(newSize: Int): LongArray {

    return withType("LongArray", arrayCopyResize(this, newSize, 0L))

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*anonymous actual fun FloatArray.copyOf(newSize: Int): FloatArray {

    return fillFrom(this, FloatArray(newSize))

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with zero values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*anonymous actual fun DoubleArray.copyOf(newSize: Int): DoubleArray {

    return fillFrom(this, DoubleArray(newSize))

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with 'false' values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with 'false' values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*anonymous actual fun CharArray.copyOf(newSize: Int): CharArray {

    return withType("CharArray", fillFrom(this, CharArray(newSize)))

* Returns new array which is a copy of the original array, resized to the given [newSize].

* The copy is either truncated or padded at the end with null char (`\u0000`) values if necessary.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\u0000`) values.

* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*anonymous actual fun <T> Array<out T>.copyOf(newSize: Int): Array<T> {

    return arrayCopyResize(this, newSize, null)

* Returns a new array which is a copy of the specified range of the original array.

* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].

* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null values.

* @sample samples.collections.Arrays.CopyOfOperations.resizingCopyOf

*anonymous actual fun <T> Array<out T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {

    return this.asDynamic().slice(fromIndex, toIndex)

* Returns a new array which is a copy of the specified range of the original array.

* - @param fromIndex the start of the range (inclusive), must be in `0..array.size`

* - @param toIndex the end of the range (exclusive), must be in `0..array.size`

* @Suppress("ACTUAL_WITHOUT_EXPECT")

*anonymous actual fun <T> Array<out T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T> {

    return this.asDynamic().slice(fromIndex, toIndex)

* Returns a new array which is a copy of the specified range of the original array.

* - @param fromIndex the start of the range (inclusive), must be in `0..array.size`

* - @param toIndex the end of the range (exclusive), must be in `0..array.size`
actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentError if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentError if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentError if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentError if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentError if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}
actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
 AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
 this.asDynamic().fill(element, fromIndex, toIndex)\n \n * @param fromIndex the start of the range (inclusive), 0 by default.\n * @param toIndex the end of the range (exclusive), size of this array by default.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgument Exception if [fromIndex] is greater than [toIndex].\n}

actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
 AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
 this.asDynamic().fill(element, fromIndex, toIndex)\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun <T> Array<out T>.plus(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun ByteArray.plus(element: Byte): ByteArray {
    return plus(byteArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun ShortArray.plus(element: Short): ShortArray {
    return plus(shortArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun IntArray.plus(element: Int): IntArray {
    return plus(intArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun LongArray.plus(element: Long): LongArray {
    return plus(longArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun FloatArray.plus(element: Float): FloatArray {
    return plus(floatArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun DoubleArray.plus(element: Double): DoubleArray {
    return plus(doubleArrayOf(element))\n \n * @returns an array containing all elements of the original array and then the given [element].\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun Collection<Byte>.plus(elements: Collection<Byte>): ByteArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n \n * @returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun Collection<Int>.plus(elements: Collection<Int>): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n \n * @returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}

actual fun Collection<Long>.plus(elements: Collection<Long>): LongArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)\n \n * @returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n}
containing all elements of the original array and then all elements of the given [elements] collection.

```kotlin
/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
public actual operator fun BooleanArray.plus(elements: Collection<Bool>): BooleanArray {
    return arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
public actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
public actual fun IntArray.sort(): Unit {
    definedExternally
}
```

Open Source Used In DNAC Platform Arica 12102
BooleanArray.toTypedArray(): Array<<Boolean>> { return js("["").slice.call(this) }\n\nCharArray.toTypedArray(): Array<Char> { return Array(size) { index -> this[index] }\n\nCopyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

`BooleanArray.toTypedArray(): Array<Boolean> {
    return js("["]").slice.call(this)
}

/**
 * Returns a typed object array containing all of the elements of this primitive array.
 * @param {Array} this
 */
public actual fun CharArray.toTypedArray(): Array<Char> {
    return Array(size) { index -> this[index] }
}

`/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
@file:kotlin.jvm.JvmName("ComparisonsKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.comparisons

/**
 * Compares two values using the specified functions [selectors] to calculate the result of the comparison.
 * The functions are called sequentially, receive the given values [a] and [b] and return [Comparable] objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do not compare as equal, the result of that comparison is returned.
 * @param selectors [array of functions]
 * @sample samples.comparisons.Comparisons.compareValuesByWithSelectors
 */
public fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>?): Int {
    require(selectors.size > 0)
    return compareValuesByImpl(a, b, selectors)
}

private fun <T> compareValuesByImpl(a: T, b: T, selectors: Array<out (T) -> Comparable<*>?>): Int {
    for (fn in selectors) {
        val v1 = fn(a)
        val v2 = fn(b)
        val diff = compareValues(v1, v2)
        if (diff != 0) return diff
    }
    return 0
}

/**
 * Compares two values using the specified [selector] function to calculate the result of the comparison.
 * The function is applied to the given values [a] and [b] and return [Comparable] objects. The result of comparison of these [Comparable] instances is returned.
 * @param selector [function]
 * @sample samples.comparisons.Comparisons.compareValuesByWithSingleSelector
 */
@kotlin.internal.InlineOnly
public inline fun <T> compareValuesBy(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValuesByImpl(a, b, selector)
}

/**
 * Compares two values using the specified [selector] function to calculate the result of the comparison.
 * The function is applied to the given values [a] and [b] and return objects of type K which are then being compared with the given [comparator].
 * @param selector [function]
 * @param comparator [comparator]
 * @sample samples.comparisons.Comparisons.compareValuesByWithComparator
 */
@kotlin.internal.InlineOnly
public inline fun <T, K> compareValuesBy(a: T, b: T, comparator: Comparator<in K>, selector: (T) -> K): Int {
    return comparator.compare(selector(a), selector(b))
}

// Not so useful without type inference for receiver of expression
//@Suppress("NOTHING_TO_INLINE")
//public inline fun <T> compareValuesWith(a: T, b: T, comparator: Comparator<T>): Int = comparator.compare(a, b)

/**
 * Compares two nullable [Comparable] values. Null is considered less than any value.
 * @param a first value
 * @param b second value
 * @sample samples.comparisons.Comparisons.compareValues
 */
public fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {
    if (a === b) return 0
    if (a == null) return -1
    if (b == null) return 1
    @Suppress("UNCHECKED_CAST")
    return (a as Comparable<Any>).compareTo(b)
}

/**
 * Creates a comparator using the sequence of functions to calculate a result of comparison.
 * The functions are called sequentially, receive the given values `a` and `b` and return [Comparable] objects. As soon as the [Comparable] instances returned by a function for `a` and `b` values do not compare as equal, the result of that comparison is returned from the [Comparator].
 * @param selectors [array of functions]
 * @sample samples.comparisons.Comparisons.compareByWithSelectors
 */
public fun <T> compareBy(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    require(selectors.size > 0)
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}

/**
 * Creates a comparator using the function to transform value to a [Comparable] instance for comparison.
 * @param selector [function]
 * @sample samples.comparisons.Comparisons.compareByWithSingleSelector
 */
@kotlin.internal.InlineOnly
public inline fun <T> compareBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        compareValuesByImpl(a, b, selector)
    }

/**
 * Creates a comparator using the specified [comparator] function to transform values being compared and then applying [selector] to compare transformed values.
 * @param comparator [function]
 * @param selector [function]
 * @sample samples.comparisons.Comparisons.compareByWithComparator
 */
@kotlin.internal.InlineOnly
public inline fun <T, K> compareBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        compareValuesByImpl(a, b, selector)
    }

Open Source Used In DNAC Platform Arica 12104
samples.comparisons.Comparisons.compareByDescendingWithSingleSelector

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> compareByDescending(crossinline selector: (T) -> Comparable<*?>?): Comparator<T> =
    Comparator { a, b -> compareValuesBy(b, a, selector) }

@kotlin.internal.InlineOnly
public inline fun <T, K> compareByDescending(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b -> compareValuesBy(b, a, comparator, selector) }

@kotlin.internal.InlineOnly
public inline fun <T> Comparator<T>.thenByDescending(crossinline selector: (T) -> Comparable<*?>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(b, a, selector)
    }

@kotlin.internal.InlineOnly
public inline fun <T, K> Comparator<T>.thenByDescending(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(b, a, comparator, selector)
    }
```

Not so useful without type inference for receiver of expression

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> thenComparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this.compare(a, b)
        if (previousCompare != 0) previousCompare else comparison(a, b)
    }
```
samples.comparisons.Comparisons.nullsFirstLastWithComparator

public fun <T : Any>
nullsFirst(comparator: Comparator<in T>): Comparator<T?> =
Comparator { a, b ->
    when {
        a
        === b -> 0
        a == null -> -1
        b == null -> 1
        else -> comparator.compare(a, b)
    }
}

/**
 * Provides a comparator of nullable [Comparable] values* considering ‘null’ value less than any other value.
 */
* @sample samples.comparisons.Comparisons.nullsFirstLastComparator

public fun <T : Any>
nullsLast(comparator: Comparator<in T>): Comparator<T?> =
Comparator { a, b ->
    when {
        a
        === b -> 0
        a == null -> 1
        b == null -> -1
        else -> comparator.compare(a, b)
    }
}

/**
 * Provides a comparator of nullable [Comparable] values* considering ‘null’ value greater than any other value.
 */
* @sample samples.comparisons.Comparisons.nullsFirstLastComparator

private class ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {
    override fun compare(a: T, b: T): Int = comparator.compare(b, a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<T> = comparator
}

private object NaturalOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = a.compareTo(b)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator
}

private object ReverseOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = b.compareTo(a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = ReverseOrderComparator
}
explaining why the implementation is missing.

```kotlin
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*

/**
 * Returns the greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Byte, b: Byte): Byte {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>()
}
```

Open Source Used In DNAC Platform Arica 12107
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a.toInt(),
    b.toInt()).unsafeCast<Short>()
}

/**
 * Returns the greater of two values.
 */
public actual fun <T : Comparable<T>> maxOf(a: T, b: T, c: T): T {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
public actual inline fun maxOf(a: Byte, b: Byte, c: Byte): Byte {
    return Math.max(a.toInt(), b.toInt(),
    c.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the greater of three values.
 */
public actual inline fun maxOf(a: Short, b: Short, c: Short): Short {
    return Math.max(a.toInt(), b.toInt(),
    c.toInt()).unsafeCast<Short>()
}

/**
 * Returns the greater of three values.
 */
public actual inline fun maxOf(a: Int, b: Int, c: Int): Int {
    return Math.max(a, b, c)
}

/**
 * Returns the greater of three values.
 */
public actual inline fun maxOf(a: Long, b: Long, c: Long): Long {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
public actual inline fun maxOf(a: Float, b: Float, c: Float): Float {
    return Math.max(a, b, c)
}

/**
 * Returns the smaller of two values.
 * If values are equal, returns the first one.
 */
public actual fun <T : Comparable<T>> minOf(a: T, b: T): T {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
public actual inline fun minOf(a: Byte, b: Byte): Byte {
    return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the smaller of two values.
 */
public actual inline fun minOf(a: Short, b: Short): Short {
    return Math.min(a.toInt(),
    b.toInt()).unsafeCast<Short>()
}

/**
 * Returns the smaller of two values.
 */
public actual inline fun minOf(a: Int, b: Int): Int {
    return Math.min(a, b)
}

/**
 * Returns the smaller of two values.
 */
public actual inline fun minOf(a: Long, b: Long): Long {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
public actual inline fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
}
inline fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
}

/**
 * Returns the smaller of three values.
 */
@.SinceKotlin("1.1")
public actual fun <T : Comparable<T>> minOf(a: T, b: T, c: T): T {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Byte, b: Byte, c: Byte): Byte {
    return Math.min(a.toInt(), b.toInt(), c.toInt().unsafeCast<Byte>().minOf())
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Short, b: Short, c: Short): Short {
    return Math.min(a.toInt(), b.toInt(), c.toInt().unsafeCast<Short>().minOf())
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Int, b: Int, c: Int): Int {
    return Math.min(a, b, c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Long, b: Long, c: Long): Long {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Float, b: Float, c: Float): Float {
    return Math.min(a, b, c)
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public actual inline fun minOf(a: Double, b: Double, c: Double): Double {
    return Math.min(a, b, c)
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//
import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns 1st *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component1(): T {
    return get(0)
}

/**
 * Returns 2nd *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component2(): T {
    return get(1)
}

/**
 * Returns 3rd *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component3(): T {
    return get(2)
}

/**
 * Returns 4th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component4(): T {
    return get(3)
}

/**
 * Returns 5th *element* from the collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> List<T>.component5(): T {
    return get(4)
}

/**
 * Returns an element at the given *index* or
 * throws an *IndexOutOfBoundsException* if the *index* is out of bounds of this collection.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
public fun <T> Iterable<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    if (this is List)
        return this.getOrElse(index, defaultValue)
    if (index < 0)
        return defaultValue(index)
    var iterator = iterator()
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return defaultValue(index)
}

/**
 * Returns an element at the given *index* or the result of calling the [defaultValue] function if the *index* is out of bounds of this collection.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrElse
 */
public actual inline fun <T> List<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T)?

samples.collections.Collections.Elements.elementAtOrElse
 */
@kotlin.internal.InlineOnly
public inline fun
<T> List<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} // Returns an element at the given [index] or 'null' if the [index] is out of bounds of this collection.

samples.collections.Collections.Elements.elementAtOrNull
 */
public fun <T> List<T>.elementAtOrNull(index: Int): T? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
} // Returns the first element matching the given [predicate], or 'null' if no such element was found.

samples.collections.Collections.Elements.getOrElse
 */
public inline fun
<T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} // Returns an element at the given [index] or 'null' if the [index] is out of bounds of this list.

samples.collections.Collections.Elements.find
 */
public inline fun <T> Iterable<T>.find(predicate: (T) -> Boolean): T? {
    return firstOrNull(predicate)
} // Returns the first element matching the given [predicate], or 'null' if no such element was found.

samples.collections.Collections.Elements.findLast
 */
public inline fun <T> List<T>.findLast(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
} // Returns the last element matching the given [predicate], or 'null' if no such element was found.

samples.collections.Collections.Elements.first
 */
public fun <T> Iterable<T>.first(): T {
    when (this) {
        is List -> return this.first()
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                throw NoSuchElementException("Collection is empty.")
            return iterator.next()
        }
    }
} // Returns the first element.

samples.collections.Collections.Elements.firstOrNull
 */
public fun <T> List<T>.firstOrNull(): T? {
    return if (isEmpty()) null else this[0]
} // Returns the first element, or 'null' if the list is empty.

samples.collections.Collections.Elements.indexOf
 */
public fun <T> Iterable<T>.indexOf(element: T): Int {
    if (this is List) return this.indexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
} // Returns first index of [element], or -1 if the collection does not contain [element].

samples.collections.Collections.Elements.indexOfFirst
 */
public inline fun <T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    return firstOrNull(predicate)
} // Returns index of the first element matching the given [predicate], or 'null' if element was not found.

samples.collections.Collections.Elements.indexOfLast
 */
public inline fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    return lastOrNull(predicate)
} // Returns index of the first element matching the given [predicate], or 'null' if element was not found.

samples.collections.Collections.Elements.getOrElse
 */
public inline fun
<T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} // Returns an element at the given [index] or 'null' if the [index] is out of bounds of this list.

samples.collections.Collections.Elements.getOrElse
 */
public inline fun
<T> List<T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} // Returns an element at the given [index] or 'null' if the [index] is out of bounds of this list.
Open Source Used In DNAC Platform Arica 12112

```kotlin
public inline fun <T> Iterable<T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return toList()
    if (n >= size) return emptyList()
    if (resultSize == 1)
        return listOf(last())
    list = ArrayList<T>(resultSize)
    if (this is List<T>) {
        if (this is RandomAccess) {
            for (index in n until size)
```
inline fun <T> List<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {
    val iterator = listIterator(size)
    while (iterator.hasPrevious()) {
        if (!predicate(iterator.previous())) {
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
            if (expectedSize == 0) return emptyList()
            return ArrayList<T>(expectedSize).apply {
                while (iterator.hasNext())
                    add(iterator.next())
            }
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given [predicate].
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun <T> Iterable<T>.takeWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Reverses elements in the list in-place.
 */
public expect fun <T> MutableList<T>.reverse(): Unit

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Iterable<T>.reversed(): List<T> {
    if (this is Collection && size <= 1) return toList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Sorts elements in the list in-place according to their natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public inline fun <T, R : Comparable<R>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareBy(selector))
}

/**
 * Sorts elements in the list in-place descending according to their natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {
    sortWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun <T : Comparable<T>> Iterable<T>.sorted(): List<T> {
    if (this is Collection) {
        if (size <= 1)
            return this.toList()<@Suppress("UNCHECKED_CAST")
        return (toTypedArray<Comparable<T>>() as Array<T>).apply { sort() }.asList()<
    } else
        return toMutableList().apply { sort() }.asList()<
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 */
public fun <T> Iterable<T>.sortedWith(comparator: Comparator<in T>): List<T> {
    if (this is Collection) {
        if (size <= 1)
            return this.toList()<@Suppress("UNCHECKED_CAST")
        return (toTypedArray<Any?>() as Array<T>).apply { sortWith(comparator) } as List<T>
    } else
        return toMutableList().apply { sortWith(comparator) }.asList()<
}

/**
 * Returns an array of Boolean containing all of the elements of this collection.
 */
public fun Collection<Boolean>.toBooleanArray(): BooleanArray {
    val result = BooleanArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Byte containing all of the elements of this collection.
 */
public fun Collection<Byte>.toByteArray(): ByteArray {
    val result = ByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of Char containing all of the elements of this collection.
 */
public fun Collection<Char>.toCharArray(): CharArray {
    val result = CharArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
elements of this collection.

```kotlin
public fun Collection<Char>.toCharArray(): CharArray {
    val result = CharArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public fun Collection<Double>.toDoubleArray(): DoubleArray {
    val result = DoubleArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public fun Collection<Float>.toFloatArray(): FloatArray {
    val result = FloatArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public fun Collection<Int>.toIntArray(): IntArray {
    val result = IntArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public fun Collection<Long>.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public fun Collection<Short>.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}
```

```kotlin
public inline fun <T, K, V> Iterable<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}
```

```kotlin
public inline fun <T, K> Iterable<T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, T>(capacity), keySelector)
}
```

```kotlin
public inline fun <T, K, V> Iterable<T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

```kotlin
public inline fun <T, K, M : MutableMap<in K, in T>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}
```

```kotlin
public inline fun <T, K, V, M : MutableMap<in K, in V>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
```

```kotlin
public inline fun <T, K, M : MutableMap<in K, in V>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
```

```kotlin
for (element in this) {
    result[index++] = element
}
return result
```
function applied to each element. If any two elements are equal, the last one gets added to the map. The returned map preserves the entry iteration order of the original collection.

```kotlin
public inline fun <K, V> Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

public inline fun <K, V, M : MutableMap<in K, in V>>
    Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

public fun <T, C : Collection<in T>>
    Iterable<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

public fun <T> Iterable<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))
}

public fun <T> Collection<T>.toMutableList(): MutableList<T> {
    return ArrayList(this)
}

public fun <T> Iterable<T>.toSet(): Set<T> {
    return toCollection(LinkedHashSet<T>(mapCapacity(size))).optimizeReadOnlySet()
}

public inline fun <T, R> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

public inline fun <T, R, C : Collection<in R>>
    Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
```

Iterable<
K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {

```kotlin
IterateT<T>.associateWithToTo(destination: C: C {
    for (item in this) {
        destination.add(item)
    }
    return destination

IterateT<T>.toCollection(destination: C: C {
    for (item in this) {
        destination.add(item)
    }
    return destination

IterateT<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))

IterateT<T>.toMutableList(): MutableList<T> {
    return ArrayList(this)

IterateT<T>.toSet(): Set<T> {
    return toCollection(LinkedHashSet<T>(mapCapacity(size))).optimizeReadOnlySet()

IterateT<T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)

IterateT<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
```
fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<T>()
        } list.add(element)
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to the element and puts to the [destination] map each group key associated with a list of corresponding values.
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<T>()
        } list.add(element)
    }
    return destination
}

/**
 * Creates a [Grouping] source from a collection to be used later with one of group-and-fold operations
 * using the specified [keySelector] function to extract a key from each element.
 */
@SinceKotlin("1.1")
public inline fun <T, K> Iterable<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object :

        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element in the original collection.
 */
public inline fun <T, R> Iterable<T>.map(transform: (T) -> R): List<R> {
    return mapTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function
 * to each element and its index in the original collection.
 */
public inline fun <T, R> Iterable<T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function
 * to each element and its index in the original collection.
 */
public inline fun <T, R : Any> Iterable<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

/**
 * Applies the given [transform] function to each element and its index in the original collection
 * and appends only the non-null results to the given [destination].
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R?): C {
    for (index, element) in this
        destination.add(transform(index, element))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original collection
 * and appends the results to the given [destination].
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapNullTo(destination: C, transform: (T) -> R?): C {
    for (element in this)
        transform?.let { destination.add(it) }
    return destination
}
transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy 
 * 
 * @param collection
 * 
 * @param transform
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<String>.transform { it.length }
 * ```
 */
public fun <T, R> Collection<T>.transform(transform: (T) -> R): Collection<R> {
    return this.toCollection(transform)
}

/**
 * Returns a collection that contains only distinct elements from the given collection.
 * 
 * @param collection
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.distinct()
 * ```
 */
public fun <T> Collection<T>.distinct(): Collection<T> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains only distinct elements from the given collection
 * 
 * @param collection
 * 
 * @param keySelector
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<String>.distinctBy { it.length }
 * ```
 */
public inline fun <T, K> Collection<T>.distinctBy(keySelector: (T) -> K): Collection<T> {
    return this.toCollection { it.key }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.intersect(Collection<Number>)
 * ```
 */
public infix fun <T> Collection<T>.intersect(other: Collection<T>): Collection<T> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in one collection
 * 
 * @param collection1
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.subtract(Collection<Number>)
 * ```
 */
public infix fun <T> Collection<T>.subtract(other: Collection<T>): Collection<T> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.toCollection { it }
 * ```
 */
public fun <T> Collection<T>.toCollection(transform: (T) -> R): Collection<R> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.intersect(Collection<Number>)
 * ```
 */
public infix fun <T> Collection<T>.intersect(other: Collection<T>): Collection<T> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.subtract(Collection<Number>)
 * ```
 */
public infix fun <T> Collection<T>.subtract(other: Collection<T>): Collection<T> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.toCollection { it }
 * ```
 */
public fun <T> Collection<T>.toCollection(transform: (T) -> R): Collection<R> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.toCollection { it }
 * ```
 */
public fun <T> Collection<T>.toCollection(transform: (T) -> R): Collection<R> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.toCollection { it }
 * ```
 */
public fun <T> Collection<T>.toCollection(transform: (T) -> R): Collection<R> {
    return this.toCollection { it }
}

/**
 * Returns a collection that contains all elements that are contained in both collections
 * 
 * @param collection1
 * 
 * @param collection2
 * 
 * @return Collection
 * 
 * Examples:
 * 
 * ```
 * Collection<Number>.toCollection { it }
 * ```
 */
public fun <T> Collection<T>.toCollection(transform: (T) -> R): Collection<R> {
    return this.toCollection { it }
}
calculates the next accumulator value.

```
@<T>.

```

```
inline fun <T> Iterable<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by 
 * the function applied to each element in the collection.
 */
public inline fun <T> Iterable<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-'null' elements, throwing an
 * [IllegalArgument] exception if there are any 'null' elements.
 */
Iterable<T>?.requireNoNulls(): Iterable<T> { for (element in this) {
    if (element == null) throw
    @Suppress("UNCHECKED_CAST")
    return this as Iterable<T>
}

@SinceKotlin("1.2")
public fun <T> Iterable<T> chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this collection into lists each not exceeding the given [size].
 * The last list in the resulting list may have less elements than the
 * given [size].
 */
@SinceKotlin("1.2")
public fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a list
 * containing all elements of the original collection except the
 * elements contained in the given [element].
 */
public operator fun <T> Iterable<T>.minus(element: T): List<T> {
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    return this.filterTo(result) { if (!removed
    \n * @param size the number of elements to take in each list, must be positive and can be greater
    than the number of elements in this collection.

@SinceKotlin("1.2")
public fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)

@SinceKotlin("1.2")
public fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)

@SinceKotlin("1.2")
public fun <T> Iterable<T> minus(elements: Array<out T>): List<T> {
    if (other.isEmpty())
        return this.toList()
    return this.filterNot { it in other }
}

@SinceKotlin("1.2")
public fun <T> Iterable<T> minus(elements: Iterable<T>): List<T> {
    val other = elements.convertToSetForSetOperationWith(this)
    return this.filterNot { it in other }
}

@SinceKotlin("1.2")
public fun <T> Iterable<T> minus(elements: Sequence<T>): List<T> {
    return this.filterNot { it in other }
}

@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.minusElement(element: T): List<T> {
return minus(element) * Splits the original collection into pair of lists. * where *first* list contains elements for which [predicate] yielded `true`, * while *second* list contains elements for which [predicate] yielded `false`. */

public inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    val first = ArrayList<T>()
    val second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Iterable<T>.plus(element: T): List<T> {
    if (this is Collection) return this.plus(element)
    val result = ArrayList<T>()
    result.addAll(this)
    result.add(element)
    return result
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Collection<T>.plus(element: T): List<T> {
    val result = ArrayList<T>(this.size + 1)
    result.addAll(this)
    result.add(element)
    return result
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Iterable<T>.plus(elements: Array<out T>): List<T> {
    if (this is Collection) return this.plus(elements)
    val result = ArrayList<T>()
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Collection<T>.plus(elements: Array<out T>): List<T> {
    if (elements is Collection) {
        val result = ArrayList<T>(this.size + elements.size)
        result.addAll(this)
        result.addAll(elements)
        return result
    } else {
        val result = ArrayList<T>(this)
        result.addAll(elements)
        return result
    }
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Iterable<T>.plus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>()
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public operator fun <T> Collection<T>.plus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>(this.size + 10)
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public Inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)
}

/**
 * Splits the original collection into pair of lists,
 * where *first* list contains elements for which [predicate]
 * yielded `true`, * while *second* list contains elements for which [predicate]
 * yielded `false`. */

public Inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)
}

/**
 * Returns a list of snapshots of the window of the given [size]
 * sliding along this collection with the given [step], where each [snapshot] is a list.
 * Several last lists may have less elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any
 * @default `false` which means partial windows won't be preserved
 * @sample samples.collections.Sequences.Transformations.takeWindows
 */

@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (windowSize < size && !partialWindows) break
            result.add(List(windowSize) { this[it + index] })
            index += step
        }
        return result
    }
}
result = ArrayList<List<T>>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
        result.add(it)
    }
    return result
}

/**
 * Returns a list of results of applying the
given [transform] function to an each list representing a view over the window of the given [size].
* Several last lists may have less elements than the given [size].
* Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
* @param size the number of elements to take in each window
* @param step the number of elements to move the window forward by on each step,
* by default 1
* @param partialWindows controls whether or not to keep partial windows in the end if any.
* by default `false` which means partial windows won't be preserved
* @sample samples.collections.Sequences.Transformations.averageWindows
 */
@SinceKotlin("1.2")
    List<R> {
        checkWindowSizeStep(size, step)
        if (this is RandomAccess && this is List) {
            val thisSize = this.size
            val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
            val result = ArrayList<R>(resultCapacity)
            val window = MovingSubList(this)
            var index = 0
            while (index in 0 until thisSize) {
                val windowSize = size.coerceAtMost(thisSize - index)
                if (!partialWindows && windowSize < size) break
                window.move(index, index + windowSize)
                result.add(transform(window))
                index += step
            }
            return result
        }
        val result = ArrayList<R>()
        windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach {
            result.add(transform(it))
        }
        return result
    }

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] array with the same index.
* The returned list has length of the shortest collection.
* @sample samples.collections.Iterables.Operations.zipIterable
 */
public infix fun <T, R> Iterable<T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` collection and the [other] collection with the same index
* using the provided [transform] function applied to each pair of elements.
* The returned list has length of the shortest collection.
* @sample samples.collections.Iterables.Operations.zipIterableWithTransform
 */
public inline fun <T, R, V> Iterable<T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {
    val first = iterator()
    val second = other.iterator()
    val list = ArrayList<V>(minOf(collectionSizeOrDefault(10), other.collectionSizeOrDefault(10)))
    while (first.hasNext() && second.hasNext()) {
        list.add(transform(first.next(), second.next()))
    }
    return list
}

/**
 * Returns a list of pairs of each two adjacent elements in this collection.
* The returned list is empty if this collection contains less than two elements.
* @sample samples.collections.Collections.Transformations.zipWithNext
 */
@SinceKotlin("1.2")
public inline fun <T, R> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> R): List<R> {
    val iterator = iterator()
    if (i >= arraySize) break
    list.add(transform(element, other[i++]))
}
Open Source Used In DNAC Platform Arica 12124

val result = mutableListOf<R>()
    while (iterator.hasNext()) {
        val next = iterator.next()
        result.add(transform(current, next))
        current = next
    }
    return result

/**
 * Appends the string from all the elements separated using [separator] and
 * using the given [prefix] and [postfix] if supplied.
 * @param limit If the collection could be huge, you can specify a non-
 * negative value of [limit], in which case only the first [limit] elements will be appended, followed by the
 * [truncated] string (which defaults to "...").
 * @param truncated Returns a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @param transform Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @param transform A \{\n * buffer.append(prefix)\n * var count = 0\n * for (element in this) {\n * if (++count > 1) buffer.append(separator)\n * if (limit < 0 || count <= limit) {\n * buffer.appendElement(element, transform)\n * else break\n * }\n * if (limit >= 0 && count > limit) buffer.append(truncated)\n * buffer.append(postfix)\n * return buffer\n * }\n * Returns this collection as an [Iterable].
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T, A : Appendable>
    Iterable<T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "\n", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) {
            buffer.appendElement(element, transform)
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Returns this collection as an [Iterable].
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Iterable<T>.asIterable(): Iterable<T> {
    return this
}

/**
 * Creates a [Sequence] instance that wraps the original collection returning
 * its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromCollection
 */
public fun <T> Iterable<T>.asSequence(): Sequence<T> {
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the
 * collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the
 * collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Iterable<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the
 * collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the
 * collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the
 * collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns a sum of all elements in the collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}
(element in this) {\n    sum += element
} \n    return sum
}\n/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Int>.sum(): Int {\n    var sum = 0\n    for (element in this) {\n        sum += element
    }\n    return sum
}\n
/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Long>.sum(): Long {\n    var sum = 0L\n    for (element in this) {\n        sum += element
    }\n    return sum
}\n
/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Float>.sum(): Float {\n    var sum = 0.0f\n    for (element in this) {\n        sum += element
    }\n    return sum
}\n
/**
 * Returns the sum of all elements in the collection.
 */
public fun Iterable<Double>.sum(): Double {\n    var sum = 0.0\n    for (element in this) {\n        sum += element
    }\n    return sum
}\n
/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.comparisons.naturalOrder
import kotlin.random.Random

/**
 * Returns the array if it's not `null`, or an empty array otherwise.
 */
public actual fun <T> Array<out T>? orEmpty(): Array<out T> = this ?: emptyArray<T>()

@JsName("copyToArrayList")
internal fun <T> Collection<T>.toTypedArray(): Array<T> = copyToArray(this)

@JsName("copyToArray")
internal fun <T> copyToArray(collection: Collection<T>): Array<T> {\n    val array = emptyArray<T>()\n    val iterator = collection.iterator()\n    while (iterator.hasNext()) {\n        array += iterator.next()\n    }\n    return array
}
internal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int, 
startIndex: Int, endIndex: Int) {
    AbstractList.checkRangeIndexes(startIndex, endIndex, source.size)
    val rangeSize = endIndex - startIndex
    AbstractList.checkRangeIndexes(destinationOffset, destination.size + rangeSize, destination.size)
    if (js("ArrayBuffer").isView(source) && 
js("ArrayBuffer").isView(destination) &&
    destination.asDynamic().set(subrange = source.asDynamic().subarray(startIndex, endIndex),
    destination.asDynamic().set(subrange, destinationOffset))
}

// no singleton map implementation in js, return map as
is/Un@Suppress("NOTHING_TO_INLINE")
internal actual inline func<K, V> Map<K, V>.toSingletonMapOrSelf(): Map<K, V> = 
this.toMutableMap()/*no@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName("CollectionsKt"*/
unpackage kotlin.collections\nun/***/
un * Returns the given iterator itself. This allows to use an instance of iterator in a "for"
loop.\nun * @sample samples.collections.Enumerators.iteratorIterator\nun */
un @sample samples.collections.Enumerators.iteratorIterator\nun */
un @kotlin.internal.InlineOnly
un public inline operator func<K, V> Iterator<K>.iterator(): Iterator<K> = this

un /**
un * Returns an iterator that wraps each element produced
un * by the original iterator into an [IndexedValue], containing the index of that element and the element itself.\nun */
un @kotlin.internal.InlineOnly
un public expect inline func<K, V> Iterator<K>.withIndex(): Iterator<IndexedValue<K>> = IndexingIterator(this)

un /**
un * Performs the given [operation] on each element of
un * this iterator.\nun */
un @sample samples.collections.Enumerators.iteratorIterator\nun */
un @kotlin.internal.InlineOnly
un public inline func<K, V> Iterator<K>.forEach(operation: (K) -> Unit): Unit {
    for (element in this) operation(element)
}

un /**
un * IndexingIterator\nun */
un internal class IndexingIterator<out K>(private val iterator: Iterator<K>) : Iterator<IndexedValue<K>>{
    private var index = 0
    final override fun hasNext(): Boolean = iterator.hasNext()
    final override fun next(): IndexedValue<K> = IndexedValue(checkIndexOverflow(index++), iterator.next())
}

un /*
un * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\nun */
un Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

un
un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: T, b: T): T

un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: Byte, b: Byte): Byte

un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: Short, b: Short): Short

un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: Int, b: Int): Int

un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: Long, b: Long): Long

un /**
un * Returns the greater of two values.\nun */
un @SinceKotlin("1.1")
un @kotlin.internal.InlineOnly
un public expect inline func<T : Comparable<T>> maxOf(a: Float, b: Float):
Float\n\n/**\n * Returns the greater of two values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Double, b: Double):
Double\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T :
Comparable<T>> maxOf(a: T, b: T, c: T): T\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Byte, b: Byte, c: Byte):
Byte\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Short, b: Short, c:
Short): Short\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Int, b: Int, c: Int):
Int\n\n/**\n * Returns the greater of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic
expect inline fun maxOf(a: Long, b: Long, c: Long): Long\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Float, b: Float, c:
Float): Float\n\n/**\n * Returns the greater of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun maxOf(a: Double, b: Double, c:
Double): Double\n\n/**\n * Returns the greater of three values according to the order specified by the given
[comparator].\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> maxOf(a: T, b: T, c: T, comparator: Comparator<in T>):
T {\n return maxOf(a, maxOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the greater of two values
according to the order specified by the given [comparator].\n * If values are equal, returns the first one.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T {\n return if
(comparator.compare(a, b) >= 0) a else b\n}\n\n/**\n * Returns the smaller of two values.\n * If values are equal,
returns the first one.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T : Comparable<T>> minOf(a: T, b: T):
T\n\n/**\n * Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic
expect inline fun minOf(a: Byte, b: Byte): Byte\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short):
Short\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int): Int\n\n/**\n
* Returns the smaller of two values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline
fun minOf(a: Long, b: Long): Long\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float):
Float\n\n/**\n * Returns the smaller of two values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double):
Double\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T :
Comparable<T>> minOf(a: T, b: T, c: T): T\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Byte, b: Byte, c: Byte):
Byte\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Short, b: Short, c:
Short): Short\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Int, b: Int, c: Int):
Int\n\n/**\n * Returns the smaller of three values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic
expect inline fun minOf(a: Long, b: Long, c: Long): Long\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Float, b: Float, c: Float):
Float\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic expect inline fun minOf(a: Double, b: Double, c:
Double): Double\n\n/**\n * Returns the smaller of three values according to the order specified by the given
[comparator].\n */\n@SinceKotlin(\"1.1\")\npublic fun <T> minOf(a: T, b: T, c: T, comparator: Comparator<in T>):
T {\n return minOf(a, minOf(b, c, comparator), comparator)\n}\n\n/**\n * Returns the smaller of two values
according to the order specified by the given [comparator].\n * If values are equal, returns the first one.\n
*/\n@SinceKotlin(\"1.1\")\npublic fun <T> minOf(a: T, b: T, comparator: Comparator<in T>): T {\n return if

Open Source Used In DNAC Platform Arica 12127


fun comparator.compare(a: Number, b: Number): Int

fun <Type> ok(to: List<Pair<K, V>>, result: List<Pair<K, V>>):

fun <Type, R> mapNotNullTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R?): List<R> {
    for (item in this) {
        transform(item)?: let { destination.add(it) }
    return destination
}

fun <Type, R> mapTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R): MutableList<R> {
    for (item in this) {
        destination.add(transform(item))
    return destination
}

fun <Type> mapNotNull(transform: (Map.Entry<K, V>) -> R?): MutableList<R> {
    for (item in this) {
        transform(item)??: let { destination.add(it) }
    return destination
}

fun <Type> mapTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R): MutableList<R> {
    for (item in this) {
        destination.add(transform(item))
    return destination
}

fun <Type, R> mapTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R): MutableList<R> {
    for (item in this) {
        destination.add(transform(item))
    return destination
}

fun <Type> mapTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R): MutableList<R> {
    for (item in this) {
        destination.add(transform(item))
    return destination
}

fun <Type> mapTo(destination: MutableList<R>, transform: (Map.Entry<K, V>) -> R): MutableList<R> {
    for (item in this) {
        destination.add(transform(item))
    return destination
}
    return
    entries.maxBy(selector)
}

/**
 * Returns the first entry having the largest value according to the provided
 * [comparator] or `null` if there are no entries.
 */
    return
    entries.minWith(comparator)
}

/**
 * Returns `true` if the map has no entries.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun <K, V> Map<out K, V>.none(): Boolean {
    return
    isEmpty()
}

/**
 * Returns `true` if no entries match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <K, V> Map<out K, V>.none(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in
        this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given [action] on each entry and
 * returns the map itself afterwards.
 */
@SinceKotlin("1.1")
public inline fun <K, V, M : Map<out K, V>>
    M.onEach(action: (Map.Entry<K, V>) -> Unit): M {
    return
    apply { for (element in
        this) action(element)
    }
}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.asIterable(): Iterable<Map.Entry<K, V>> {
    return
    entries
}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries
 * when being iterated.
 */
public fun <K, V> Map<out K, V>.asSequence(): Sequence<Map.Entry<K, V>> {
    return
    entries.asSequence()
}

package

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

package

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns
 * `true` if 
 * [element] is found in the sequence.
 * 
 * The operation is _terminal_.
 */
public operator fun
    <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return
    indexOf(element)
        >= 0
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the
 * [index] is out of bounds of this sequence.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
public fun <T> Sequence<T>.elementAt(index: Int): T {
    return
    elementAtOrElse(index) {
        throw IndexOutOfBoundsException("Sequence doesn't contain element at index $index.")
    }
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
 * [index] is out of bounds of this sequence.
 * 
 * The operation is _terminal_.
 */
public inline fun <T>
    Sequence<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    if (index < 0)
        return
        defaultValue(index)
    val iterator = iterator() as
        IndexOutOfBoundsException("Sequence doesn't contain element at index $index")

    while (iterator.hasNext())
        if (index == iterator.next() + 0)
            return
            defaultValue(iterator.next() + 1)

    return
    defaultValue(index)
}

/**
 * Converts this sequence to a floating-point range.
 * 
 * @sample samples.collections.Collections.Aggregates.toBeSequence
 */
public fun <T> Sequence<T>.rangeTo(what: Float): ClosedFloatingPointRange<T> {
    return
    this.toDouble().rangeTo(what.toDouble()).unsafeCast<ClosedFloatingPointRange<T>>()
}

/**
 * Returns
 * `true` if the range contains [element].
 * 
 * The operation is _terminal_.
 */
public operator fun
    <@kotlin.internal.OnlyInputTypes T> ClosedFloatingPointRange<T>.contains(element: T): Boolean {
    return
    indexOfElement(element)
        >= 0
}

/**
 * Returns an element at the given [index].
 * 
 * The operation is _terminal_.
 */
public fun <T> ClosedFloatingPointRange<T>.getElementAt(index: Int): T {
    return
    this[index]
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
Sequence<T>.elementAtOrNull(index: Int): T? {
    if (index < 0) return null
    var iterator = iterator()
    var element = iterator.next()?
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return null
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.find(predicate: (T) -> Boolean): T? {
    return firstOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.findLast(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public fun <T> Sequence<T>.first(): T {
    val iterator = iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Sequence is empty.")
    return iterator.next()
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun <T> Sequence<T>.first(predicate: (T) -> Boolean): T {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Sequence contains no element matching the predicate.")
}

/**
 * Returns first element, or `null` if the sequence is empty.
 */
public fun <T> Sequence<T>.firstOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    return iterator.next()
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> Sequence<T>.firstOrNull(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first index of [element], or -1 if the sequence does not contain such element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.indexOf(element: T): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the sequence does not contain such element.
 */
public inline fun <T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the sequence does not contain such element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last element, or `null` if the sequence is empty.
 */
public fun <T> Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}
Sequence<T>.filterIsInstanceTo(destination: C): C { 
    return destination.add(element) 
} // Appends all elements that are not 'null'

Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> { 
    return FilteringSequence(this, false, predicate) 
} // Filters out elements that do not match the given predicate

Sequence<T>.filterNullTo(destination: C): C {
    for (element in this) if (element is R) destination.add(element)
    return destination 
} // Appends all elements that match the given predicate

Sequence<T>.filterNotNull(): Sequence<T> {
    return filterNot { it == null } 
} // Returns a sequence containing all elements not null

Sequence<T>.filterNotNull(): Sequence<T> {
    return filterNot { it == null } 
} // Returns a sequence containing all elements not null

Sequence<T>.filterNotNullTo(destination: C): C {
    for (element in this) if (element != null) destination.add(element)
    return destination 
} // Appends all elements not matching the given predicate

Sequence<T>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination 
} // Filters out elements that match the given predicate

Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination 
} // Filters in elements that match the given predicate

Sequence<T>.take(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return when {
        n == 0 -> emptySequence()
        n < this @DropTakeSequence.dropSize -> this.take(n)
        else -> TakeSequence(this, n)
    }
} // Returns a sequence containing first n elements

Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {
    return TakeWhileSequence(this, predicate)
} // Returns a sequence containing first elements satisfying the given predicate

Sequence<T>.sorted(): Sequence<T> { 
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this@sorted.toMutableList()
            sortedList.sort()
            return sortedList.iterator()
        }
    }
} // Returns a sequence sorted according to their natural sort order

Sequence<T>.sortedBy(crossinline selector: (T) -> R?): Sequence<T> { 
    return sortedWith(compareBy(selector)) 
} // Returns a sequence sorted according to natural sort order of the value returned by specified selector

Sequence<T>.sortedByDescending(crossinline selector: (T) -> R?): Sequence<T> { 
    return sortedWith(compareByDescending(selector)) 
} // Returns a sequence sorted according to natural sort order of the value returned by specified selector

Sequence<T>.sortedDescending(): Sequence<T> { 
    return sortedWith(reverseOrder()) 
} // Returns a sequence sorted according to reverse order

Sequence<T>.sortedBy(crossinline selector: (T) -> R?): Sequence<T> { 
    return sortedWith(compareBy(selector)) 
} // Returns a sequence sorted according to natural sort order of the value returned by specified selector

Sequence<T>.sortedByDescending(crossinline selector: (T) -> R?): Sequence<T> { 
    return sortedWith(compareByDescending(selector)) 
} // Returns a sequence sorted according to natural sort order of the value returned by specified selector

Sequence<T>.sortedDescending(): Sequence<T> { 
    return sortedWith(reverseOrder()) 
} // Returns a sequence sorted according to reverse order
override fun iterator(): Iterator<T> {
    val sortedList = this@sortedWith.toMutableList()
    sortedList.sortWith(comparator)
    return sortedList.iterator()
}

/**
 * Returns a 
[Map] containing key-value pairs provided by [transform] function\n* applied to elements of the given sequence.\n* \n* If any of two pairs would have the same key the last one gets added to the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
public inline fun <T, K, V> Sequence<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    return associateTo(LinkedHashMap<K, V>(), transform)\n    }
}

/**
 * Returns a 
[Map] containing key-value pairs provided by [keySelector] function applied to each element.\n* \n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
public inline fun <T, K> Sequence<T>.associateBy(keySelector: (T) -> K): Map<K, T> {
    return associateByTo(LinkedHashMap<K, T>(), keySelector)\n    }
}

/**
 * Returns a 
[Map] containing the elements from the given sequence indexed by the key returned from [keySelector] function applied to each element.\n* \n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
public inline fun <T, K, V> Sequence<T>.associateByTo(destination: MutableList<K, V>, keySelector: (T) -> K): MutableList<K, V> {
    for (element in this) {
        destination.put(keySelector(element))\n    }
    return destination\n    }
}

/**
 * Returns a 
[Map] where keys are elements from the given sequence and values are provided by \n[transform] function applied to each element.\n* \n* If any two elements are equal, the last one overwrites the former value in the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
@SinceKotlin("1.3")
public inline fun <K, V> Sequence<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>()\n    return associateWithTo(result, valueSelector)\n    }
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n* where key is provided by the [keySelector] function applied to each element of the given sequence.\n* \n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<in K, in V>> Sequence<K>.associateWithTo(destination: M, keySelector: (K) -> V): M {
    for (element in this) {
        destination.put(keySelector(element))\n    }
    return destination\n    }
}

/**
 * Returns a 
[Map] where keys are elements from the given sequence and values are produced by [valueSelector] function\n* applied to each element.\n* \n* If any two elements are equal, the last one overwrites the former value in the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
@SinceKotlin("1.3")
public inline fun <K, V> Sequence<K>.associateWithTo(destination: MutableList<K, V>, valueSelector: (K) -> V): MutableList<K, V> {
    for (element in this) {
        destination.put(valueSelector(element))\n    }
    return destination\n    }
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs for each element of the given sequence,\n* where key is the element itself and value is provided by [valueSelector] function applied to that key.\n* \n* If any two elements are equal, the last one overwrites the former value in the map.\n* \nThe returned map preserves the entry iteration order of the original sequence.\n* \nThe operation is _terminal_.\n*/
@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<in K, in V>> Sequence<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(valueSelector(element))\n    }
    return destination\n    }
}

/**
 * Appends all elements to the given 
[destination] collection.\n* \nThe operation is _terminal_.\n*/
public fun <T, C : MutableCollection<in T>> Sequence<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.put(item)\n    }
    return destination\n    }
}
Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function\n * to each element in the original sequence.\n * The operation is _intermediate_ and _stateless_.\n * @param \[transform\] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n * @param \[destination\] collection to which the result elements are added.\n * @param \[comparator\] function used to remove duplicates.\n * @param \[keySelector\] function used to generate the keys for sorting.\n * The operation is _stateful_.\n */
public fun <T, R : Any> Grouping<T, R>.distinctBy(selector: (T) -> R): Sequence<R> {
    return TransformingIndexedSequence(this, transform).mapNotNull(transformed =>
        selector(transformed))
            .distinctBy
            .transformed
            .sortedBy
            .terminal
            .to(destination)
}

/**
 * Applies the given \[transform\] function to each element and its index in the original sequence\n * and appends only the non-null results to the given \[destination\].\n * The operation is _terminal_.\n */
public inline fun <T, R, C : MutableCollection<in R>> Grouping<T, R>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    for (element in this)
        destination.add(transform(checkIndexOverflow(index++), element))
    return destination
}

/**
 * Returns a sequence containing only the non-null results of applying the given \[transform\] function\n * to each element in the original sequence.\n * The operation is _intermediate_ and _stateless_.\n * @param \[transform\] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n * @param \[destination\] collection to which the result elements are added.\n * @param \[comparator\] function used to remove duplicates.\n * @param \[keySelector\] function used to generate the keys for sorting.\n * The operation is _stateful_.\n */
public fun <T, R : Any> Grouping<T, R>.distinctBy(selector: (T) -> R): Sequence<R> {
    return TransformingIndexedSequence(this, transform).mapNotNull(transformed =>
        selector(transformed))
            .distinctBy
            .transformed
            .sortedBy
            .terminal
            .to(destination)
}

/**
 * Applies the given \[transform\] function to each element and its index in the original sequence\n * and appends only the non-null results to the given \[destination\].\n * The operation is _terminal_.\n */
public inline fun <T, R, C : MutableCollection<in R>> Grouping<T, R>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    for (element in this)
        destination.add(transform(checkIndexOverflow(index++), element))
    return destination
}

/**
 * Returns a sequence containing only the non-null results of applying the given \[transform\] function\n * to each element in the original sequence.\n * The operation is _intermediate_ and _stateless_.\n * @param \[transform\] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n * @param \[destination\] collection to which the result elements are added.\n * @param \[comparator\] function used to remove duplicates.\n * @param \[keySelector\] function used to generate the keys for sorting.\n * The operation is _stateful_.\n */
public fun <T, R : Any> Grouping<T, R>.distinctBy(selector: (T) -> R): Sequence<R> {
    return TransformingIndexedSequence(this, transform).mapNotNull(transformed =>
        selector(transformed))
            .distinctBy
            .transformed
            .sortedBy
            .terminal
            .to(destination)
}

/**
 * Applies the given \[transform\] function to each element of the original sequence
 * and appends the results to the given \[destination\].\n * The operation is _terminal_.\n */
public inline fun <T, R, C : MutableCollection<in R>> Grouping<T, R>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
* Returns a mutable set containing all distinct elements from the given sequence.
  The returned set preserves the element iteration order of the original sequence.
  The operation is _terminal_ *The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {
    val set = LinkedHashSet<T>()
    for (item in this)
        set.add(item)
    return set
}
```

* Returns `true` if all elements match the given [predicate].
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}
```

* Returns `true` if at least one element matches the given [predicate].
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T> Sequence<T>.any(): Boolean {
    return iterator().hasNext()
}
```

* Returns `true` if at least one element matches the given [predicate].
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public inline fun <T> Sequence<T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
```

* Returns the number of elements in this sequence.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T> Sequence<T>.count(): Int {
    var count = 0
    for (element in this)
        checkCountOverflow(++count)
    return count
}
```

* Returns the number of elements matching the given [predicate].
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public inline fun <T> Sequence<T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) checkCountOverflow(++count)
    return count
}
```

* Performs the given [action] on each element.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public inline fun <T> Sequence<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}
```

* Performs the given [action] on each element, providing sequential index with the element.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public inline fun <T> Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}
```

* Returns the largest element or `null` if there are no elements.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
@SinceKotlin("1.1")
public fun Sequence<Double>.max(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    if (max.isNaN()) return max
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}
```

* Returns the largest element or `null` if there are no elements.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
@SinceKotlin("1.1")
public fun Sequence<Float>.max(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    if (max.isNaN()) return max
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}
```

* Returns the largest element or `null` if there are no elements.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T : Comparable<T>> Sequence<T>.max(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e == null) return e
        if (max < e) max = e
    }
    return max
}
```

* Returns the largest element or `null` if there are no elements.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T : Comparable<T>> Sequence<T>.max(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}
```

* Returns the first element yielding the largest value of the given function or `null` if there are no elements.
  The operation is _terminal_ *The operation is _final_.*

```kotlin
public fun <T, R : Comparable<R>> Sequence<T, R>.maxBy(on: (T) -> R): R? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}
```
Sequence<T>.map { action(it) } Returns the elements mapped by the provided 
operation according to the provided 
operation.
* The operation is _intermediate_.
* @SinceKotlin("1.1")
public fun <T> Sequence<T>.map(selector: (T) -> R): Sequence<R> {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var elem = iterator.next()
    while (iterator.hasNext()) {
        val e = selector(elem)
        elem = e
    }
    return iterator
}
Open Source Used In DNAC Platform Arica 12138

```kotlin
// Terminal
public inline fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

// Terminal
public inline fun <S, T> Sequence<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

// Terminal
public inline fun <S, T> Sequence<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

// Intermediate and Stateful
public fun <T : Any> Sequence<T?>.requireNoNulls(): Sequence<T> {
    return map { it ?: throw IllegalArgumentException("null element found in $this.") }
}

// Intermediate and Stateful
@SinceKotlin("1.2")
public fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {
    return windowed(size, size, partialWindows = true)
}

// Intermediate and Stateful
@SinceKotlin("1.2")
public fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

// Intermediate
public operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            var removed = false
            return this@minus.filter { if (!removed && it == element) { removed = true; false } else true }.iterator()
        }
    }
}

// Intermediate
public operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {
    if (elements.isEmpty()) return this
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            return this@minus.filterNot { it in other }.iterator()
        }
    }
}

// Intermediate
public operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {
    if (elements.isEmpty()) return this
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            return this@minus.filterNot { it in other }.iterator()
        }
    }
}
```

Sequence<T> { 
    return object: Sequence<T> { 
        override fun iterator(): Iterator<T> { 
            val other = elements.convertToSetForSetOperation() 
            if (other.isEmpty()) return this@minus.iterator() 
            else return this@minus.filterNot { it in other }.iterator() 
        } 
    } 
} 

/** 
 * Returns a sequence containing all elements of original sequence except the elements contained in the given [elements] sequence.
 * 
 * Note that the source sequence and the sequence being subtracted are iterated only when an `iterator` is requested
 * from it * the resulting sequence. Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ for this sequence and _terminal_ and _stateful_ for the [elements] sequence.
 * 
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a * correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 */
public operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> { 
    return object: Sequence<T> { 
        override fun iterator(): Iterator<T> { 
            val other = elements.toHashSet() 
            if (other.isEmpty()) return this@minus.iterator() 
            else return this@minus.filterNot { it in other }.iterator() 
        } 
    } 
}

/** 
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> { 
    return minus(element) 
}

/** 
 * Splits the original sequence into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, *second* list contains elements for which [predicate] yielded `false`.
 * 
 * The operation is _terminal_ and _stateless_.
 * 
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a * correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 */
public inline fun <T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> { 
    val first = ArrayList<T>() 
    val second = ArrayList<T>() 
    for (element in this) {
        if (predicate(element)) {
            first.add(element) 
        } else {
            second.add(element) 
        }
    }
    return Pair(first, second) 
}

/** 
 * Returns a sequence containing all elements of original sequence and then the given [element].
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> { 
    return plus(element) 
}

/** 
 * Returns a sequence of snapshots of the window of the given [size] sliding along this sequence with the given [step], where each snapshot is a list.
 * 
 * Several last lists may have less elements than the given [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * 
 * The operation is _intermediate_ and _stateless_.
 * 
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a * correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 */
public operator fun <T> Sequence<T>.plus(elements: Array<out T>): Sequence<T> { 
    return this.plus(elements.asList()) 
}

/** 
 * Returns a sequence of snapshots of the window of the given [elements] collection sliding along this sequence with the given [size] and [step], where each snapshot is a list.
 * 
 * Several last lists may have less elements than the given [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * 
 * The operation is _intermediate_ and _stateless_.
 * 
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a * correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 */
public operator fun <T> Sequence<T>.plus(elements: Iterable<T>): Sequence<T> { 
    return sequenceOf(this, elements.asSequence()).flatten() 
}

/** 
 * Returns a sequence of snapshots of the window of the given [elements] sequence sliding along this sequence with the given [size] and [step], where each snapshot is a list.
 * 
 * Several last lists may have less elements than the given [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * 
 * The operation is _intermediate_ and _stateless_.
 * 
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a * correct and stable implementation of `hashCode()` that doesn’t change between successive invocations.
 */
public operator fun <T> Sequence<T>.plus(elements: Sequence<T>): Sequence<T> { 
    return sequenceOf(this, elements).flatten() 
}

/** 
 * Returns a sequence containing all elements of original sequence and then the given [element].
 * 
 * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from it * the resulting sequence.
 * 
 * Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
public operator fun <T> Sequence<T>.plusArray(array: Array<out T>): Sequence<T> { 
    return this.plus(array.asList()) 
}

/** 
 * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] array.
 * 
 * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from it * the resulting sequence.
 * 
 * Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusArrayElement(element: T): Sequence<T> { 
    return plusArray(element) 
}

/** 
 * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] collection.
 * 
 * Note that the source sequence and the collection being added are iterated only when an `iterator` is requested from it * the resulting sequence.
 * 
 * Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusCollection(collection: Collection<out T>): Sequence<T> { 
    return plusCollection(collection.asSequence()) 
}

/** 
 * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] sequence.
 * 
 * Note that the source sequence and the sequence being added are iterated only when an `iterator` is requested from it * the resulting sequence.
 * 
 * Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusSequence(sequence: Sequence<out T>): Sequence<T> { 
    return plusSequence(sequence) 
}

/** 
 * Returns a sequence containing all elements of original sequence and then all elements of the given [elements] array.
 * 
 * Note that the source sequence and the array being added are iterated only when an `iterator` is requested from it * the resulting sequence.
 * 
 * Changing any of them between successive calls to `iterator` may affect the result.
 * 
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusSequenceElement(element: T): Sequence<T> { 
    return plusSequence(element) 
}
windows won't be preserved

@sample samples.collections.Sequences.Transformations.takeWindows

*/
@SinceKotlin("1.2")
    return windowedSequence(size, step, partialWindows, reuseBuffer = true).map(transform) + 
}

/**
 * Returns a sequence of results of applying the given [transform] function to an each list representing a view over the window of the given [size].
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * Several last lists may have less elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * The resulting sequence ends as soon as the shortest input sequence ends.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Sequences.Transformations.zipWithNext
 */
@SinceKotlin("1.2")
public fun <T, V> Sequence<T>.zipWithNext(transform: (a: T, b: T) -> V): Sequence<V> {
    var current = iterator.next()
    while (iterator.hasNext()) {
        val next = iterator.next()
        yield(transform(current, next))
        current = next
    }
    yield(transform(transform, current))
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * The operation is _terminal_.
 * @sample sample.collections.Collections.Transformations.joinTo
 */
Sequence<T>.joinTo(buffer: Appendable, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null) { A : Appendable
    A +
}
If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to `"..."`).

The operation is _terminal_.

```kotlin
public fun <T> Sequence<T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = ", ", limit: Int = -1, truncated: CharSequence = "...") String {
    return joinToString(StringBuilder(), separator, prefix, postfix, limit, truncated).toString()
}
```

```kotlin
/**
 * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.
 */
public fun <T> Sequence<T>.asIterable(): Iterable<T> {
    return Iterable { this.iterator() }
}
```

```kotlin
/**
 * Returns this sequence as a [Sequence].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.asSequence(): Sequence<T> {
    return this
}
```

```kotlin
/**
 * Returns an average value of elements in the sequence. The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Sequence<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
/**
 * Returns an average value of elements in the sequence. The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Sequence<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
/**
 * Returns an average value of elements in the sequence. The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Sequence<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
/**
 * Returns an average value of elements in the sequence. The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Sequence<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
```

```kotlin
/**
 * Returns an average value of elements in the sequence. The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Sequence<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}
```
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the LICENSE.txt file.
 */

package kotlin.collections

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns a set containing all elements of the original set except the given 
 * [element].
 * The returned set preserves the element iteration order of the original set.
 * Doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(element: T): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(size))
    var removed = false
    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given 
 * [elements] array.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
 * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    val result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given 
 * [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have
 * a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    val result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the given 
 * [element].
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given 
 * [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(element: T): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(size + 1))
    result.addAll(this)
    result.add(element)
    return result
}

/**
 * Returns a set containing all elements of the original set and the given 
 * [elements] array, which aren't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * Doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(this.size + elements.size))
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and the given 
 * [elements] sequence, which aren't already in this set.
 * The returned set preserves the element iteration order of the original set.
 * Doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(this.size + 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}

/*
 */

"Open Source Used In DNAC Platform Arica 12142"
the original set and then the given [element] if it isn't already in this set.

* The returned set preserves the element iteration order of the original set.

* Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.

* Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.

* Returns the first character matching the given [predicate], or `null` if no such character was found.

* Returns the first character matching the given [predicate].

* Returns index of the first character matching the given [predicate], or -1 if the char sequence does not contain such character.

* Returns index of the first character matching the given [predicate] or `-1` if the char sequence is empty.

* Returns the last character matching the given [predicate], or `null` if no such character was found.

* Returns the last character matching the given [predicate].

* Returns index of the last character matching the given [predicate], or -1 if the char sequence does not contain such character.

* Returns index of the last character matching the given [predicate] or `-1` if the char sequence is empty.

* Returns the first character matching the given [predicate], or `null` if the [index] is out of bounds of this char sequence.

* Returns the first character matching the given [predicate] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.

* Returns the first character matching the given [predicate] or `null` if character was not found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.

* Returns the first character matching the given [predicate] or `null` if no such character was found.
CharSequence.last(predicate: (Char) -> Boolean): Char {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if (predicate(element)) return element\n    }\n    throw NoSuchElementException("Char sequence contains no character matching the predicate.\")\n}\n
*\n
public fun CharSequence.lastOrNull(): Char? {\n    return if (isEmpty()) null else this[length - 1]\n}\n
/**\n * Returns the last character, or `null` if the\n * char sequence is empty.\n */\npublic fun CharSequence.lastOrNull(predicate: (Char) -> Boolean): Char? {\n    for (index in this.indices.reversed()) {\n        val element = this[index]\n        if (predicate(element)) return element\n    }\n    return null\n}\n
*\n
public fun CharSequence.random(): Char {\n    return random(Random)\n}\n
/**\n * Returns a random character from this char sequence using the specified\n * source of randomness.\n */\npublic fun CharSequence.random(random: Random): Char {\n    if (isEmpty())\n        throw NoSuchElementException("Char sequence is empty.\")\n    return get(random.nextInt(length))\n}\n
/**\n * Returns a single character, or throws an exception if the char sequence is empty or has more than one character.\n */\npublic fun CharSequence.single(): Char {\n    return when (length) {\n        0 -> throw NoSuchElementException("Char sequence is empty.\")\n        1 -> this[0]\n        else -> throw IllegalArgumentException("Char sequence contains more than one element.\")\n    }\n}\n
/**\n * Returns the single character matching the given [predicate], or throws an exception if there is no or more than one matching character.\n */\npublic inline fun CharSequence.singleOrNull(predicate: (Char) -> Boolean): Char? {\n    var single: Char? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException("Char sequence contains more than one matching element.\")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException("Char sequence contains no character matching the predicate.\")\n    return single\n}\n
/**\n * Returns single character, or `null` if the char sequence is empty or has more than one character.\n */\npublic fun CharSequence.singleOrNull(): Char? {\n    return if (length == 1) this[0] else null\n}\n
/**\n * Returns a subsequence of this char sequence with the first [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n */\npublic fun CharSequence.drop(n: Int): CharSequence {\n    require(n >= 0) { "Requested character count $n is less than zero.\" }\n    return subSequence(n.coerceAtMost(length), length)\n}\n
/**\n * Returns a string with the first [n] characters removed.\n * \n * @throws IllegalArgumentException if [n] is negative.\n */\npublic fun String.drop(n: Int): String {\n    require(n >= 0) { "Requested character count $n is less than zero.\" }\n    return substring(n.coerceAtMost(length))\n}\n
/**\n * Returns a subsequence of this char sequence containing all\n * characters except last characters that satisfy the given [predicate].\n */\npublic inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {\n    for (index in lastIndex downTo 0) {\n        if (!predicate(this[index])) return subSequence(0, index + 1)\n    }\n    return this\n}\n
/**\n * Returns a string containing all characters except last characters that satisfy the given [predicate].\n */\n
```kotlin
/**
 * Returns a subsequence of this char sequence containing all
 * characters except first characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return substring(0, index + 1)
    return ""
}

/**
 * Returns a string containing only those characters from the original char sequence that
 * match the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun CharSequence.dropWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return ""
}

/**
 * Returns a string containing all characters except first characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropWhile(predicate: (Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original string that match the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself and returns the result of predicate evaluation on the character.
 * @sample samples.text.Strings.drop
 */
public inline fun CharSequence.dropWhileIndexed(predicate: (Int, Char) -> Boolean): CharSequence {
    return filterIndexedTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing only those characters from the original string that do not match the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself and returns the result of predicate evaluation on the character.
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropWhileIndexed(predicate: (Int, Char) -> Boolean): String {
    return filterIndexedTo(StringBuilder(), predicate).toString()
}

/**
 * Appends all characters matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of a character and the character itself and returns the result of predicate evaluation on the character.
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropWhileIndexedTo(destination: Appendable, predicate: (Int, Char) -> Boolean): Appendable {
    for (index in 0 until length) {
        val element = get(index)
        if (predicate(index, element)) destination.append(element)
    }
    return destination
}

/**
 * Application of the given function to the specified range of indices.
 * @sample samples.text.Strings.drop
 */
public fun CharSequence.slice(indices: IntRange): CharSequence {
    if (indices.isEmpty()) return ""
    return subSequence(indices)
}

/**
 * Application of the given function to the specified range of indices.
 * @sample samples.text.Strings.drop
 */
public fun String.slice(indices: IntRange): String {
    if (indices.isEmpty()) return ""
    return substring(indices)
}

/**
 * Application of the given function to the specified range of indices.
 * @sample samples.text.Strings.drop
 */
public fun CharSequence.slice(indices: Iterable<Int>): CharSequence {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return ""
    val result = StringBuilder(size)
    for (i in indices) {
        result.append(get(i))
    }
    return result
}

/**
 * Application of the given function to the specified range of indices.
 * @sample samples.text.Strings.drop
 */
public fun String.slice(indices: Iterable<Int>): String {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return ""
    return result.append(get(i))
}
```

---

Open Source Used In DNAC Platform Arica  12145
CharSequence).slice(indices).toString() * Returns a subsequence of this char sequence containing the first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.

@throws IllegalArgumentException if [n] is negative.

* 
* * @sample samples.text.Strings.take

CharSequence.take(n: Int): CharSequence { 
    require(n >= 0) { "Requested character count $n is less than zero.\" "}
    return subSequence(0, n.coerceAtMost(length))
} * Returns a string containing the first [n] characters from this string, or the entire string if this string is shorter.

* 
* * @throws IllegalArgumentException if [n] is negative.

* 
* * @sample samples.text.Strings.take

CharSequence.takeLast(n: Int): CharSequence { 
    require(n >= 0) { "Requested character count $n is less than zero.\" "}
    return subSequence(length - n.coerceAtMost(length), length)
} * Returns a string containing the last [n] characters from this string, or the entire string if this string is shorter.

* 
* * @throws IllegalArgumentException if [n] is negative.

* 
* * @sample samples.text.Strings.take

CharSequence.takeLast(n: Int): String { 
    require(n >= 0) { "Requested character count $n is less than zero.\" "}
    return substring(length - n.coerceAtMost(length))
} * Returns a subsequence of this char sequence containing the last [n] characters that satisfy the given [predicate].

* 
* * @sample samples.text.Strings.take

CharSequence.takeLastWhile(predicate: (Char) -> Boolean): CharSequence { 
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return subSequence(index + 1, length)
        }
    }
    return subSequence(0, length)
} * Returns a string containing last characters that satisfy the given [predicate].

* 
* * @sample samples.text.Strings.take

CharSequence.takeLastWhile(predicate: (Char) -> Boolean): String { 
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return substring(index + 1)
        }
    }
    return this
} * Returns a string containing last characters that satisfy the given [predicate].

* 
* * @sample samples.text.Strings.take

CharSequence.reverse(): CharSequence { 
    return StringBuilder(this).reverse()
} * Returns a char sequence with characters in reversed order.

* 
* * @sample samples.text.Strings.take

String.reverse(): String { 
    return this
} * Returns a string with characters in reversed order.

* 
* * @kotlin.internal.InlineOnly

CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> { 
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
} * Returns a [Map] containing key-value pairs provided by [transform] function applied to characters of the given char sequence.

* 
* * If any of two pairs would have the same key the last one gets added to the map.

* 
* * @Return The returned map preserves the entry iteration order of the original char sequence.

* 
* * @sample maps.text.Strings.associate

String.associateBy(keySelector: (Char) -> K): Map<K, Char> { 
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Char>(capacity), keySelector)
} * Returns a [Map] containing the characters from the given char sequence indexed by the key returned from [keySelector] function applied to each character.

* 
* * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.

* 
* * The returned map preserves the entry iteration order of the original char sequence.

* 
* * @sample maps.text.Strings.associateBy

CharSequence.associateBy(keySelector: (Char) -> K): <K, V> { 
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector)
} * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to characters of the given char sequence.

* 
* * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.

* 
* * The returned map preserves the
entry iteration order of the original char sequence.
CharSequence.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
    keySelector, valueTransform)
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is
 * provided by the [keySelector] function applied to each character of the given char sequence.
 * Any two characters would have the same key returned by [keySelector] the last one gets added to the map.
 * @sample samples.text.Strings.associateWith
 */
@SinceKotlin("1.3")
public inline fun <V> CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V> {
    val result = LinkedHashMap<Char, V>(mapCapacity(length).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

/**
 * Appends all characters to the given [destination] collection.
 */
public fun <C : MutableCollection<in Char>> CharSequence.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each character of original char sequence.
 * @sample samples.text.Strings.flatMapTo
 */
public inline fun <R> CharSequence.flatMapTo(destination: C, transform: (Char) -> Array<R>?): List<R> {
    val list = transformTo(destination)
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each character of original char sequence.
 * @sample samples.text.Strings.flatMap
 */
public inline fun <R> CharSequence.flatMapTo(destination: C, transform: (Char) -> R): List<R> {
    return flatMapTo(destination)
}
*Groups characters of the original char sequence by the key returned by the given [keySelector] function* applied to each character and returns a map where each group key is associated with a list of corresponding characters. The returned map preserves the entry iteration order of the keys produced from the original char sequence.

```kotlin
/**
 * Groups characters of the original char sequence by the key returned by the given [keySelector] function applied to each character and returns a map where each group key is associated with a list of corresponding characters. The returned map preserves the entry iteration order of the keys produced from the original char sequence.
 */
public inline fun <K> CharSequence.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {
    return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)
}

/**
 * Returns a list containing the results of applying the given [transform] function * to each character and its index in the original char sequence * and returns a list containing the results of applying the given [transform] function * to each character and its index in the original char sequence.
 */
public inline fun <K : MutableCollection<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (Int, Char) -> R): List<R> {  
    return mapIndexedNotNullTo(ArrayList<R>(), transform, destination)
}
```

---

It seems there's a formatting issue with the text, possibly indicating a page break or a continuation of the same passage from the previous page. The text above is a continuation of the one provided, focusing on functionalities related to grouping and mapping operations on sequences of characters.
R?): C \{ \n    forEachIndexed { index, element -> transform(index, element)?\.let { destination\.add(it) } }\n    return destination
}\n\n* @param [transform] function that takes the index of a character and the character itself * and returns the result of the transform applied to the character.

* @param [transform] function that returns the result of applying the given [transform] function to each character in the original char sequence.

* Returns a list containing only the non-null results of applying the given [transform] function to each character in the original char sequence.

* Returns the number of characters matching the given [predicate].

* Returns the length of this char sequence.

* Returns `true` if all characters match the given [predicate].

* Returns `true` if char sequence has at least one character.

* Returns `true` if at least one character matches the given [predicate].

* Returns value starting with [initial] value and applying [operation] from left to right to current accumulator value and each character.

* Returns value starting with [initial] value and applying [operation] from right to left to current accumulator value and each character with its index in the original char sequence.

* Returns a lazy [Iterable] that wraps each character of the original char sequence * and appends the results to the given [destination].

* Returns a list containing only the non-null results of applying the given [transform] function to each character in the original char sequence.

* Returns a list containing only the non-null results of applying the given [transform] function to each character in the original char sequence.

* Returns value starting with [initial] value and applying [operation] from left to right to current accumulator value and each character.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.

* Returns value starting with [initial] value and applying [operation] from right to left to each character and current accumulator value.
Open Source Used In DNAC Platform Arica  12150

```kotlin
(accumulator = operation(index, get(index), accumulator))

/* *public inline fun CharSequence.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}*/

(*@param [action] that takes the index of a character and the character itself)*

/* perform the given [action] on each character */

/* *public inline fun CharSequence.forEachIndexed(action: (Int, Char) -> Unit): Unit {
    for (index in 1..lastIndex) action(index, this[index])
}*/

/* *public fun CharSequence.reduce(operation: (Char, Char) -> Char): Char {
    var accumulator = this[0]
    for (index in 1..lastIndex) accumulator = operation(accumulator, this[index])
    return accumulator
}*/

/* perform the given [operation] on each character, sequential index with the character.

/* perform the given [operation] on each character, with its index in the original char sequence.

/* *public fun CharSequence.reduceIndexed(operation: (Int, Int, Char) -> Int): Int {
    var accumulator = 0
    for (index in 1..lastIndex) accumulator = operation(accumulator, index, this[index])
    return accumulator
}*/
```

Open Source Used In DNAC Platform Arica  12150
operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with last character and applying [operation] from right to left to each character and current accumulator value.
 */
public inline fun CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last character and applying [operation] from right to left to each character with its index in the original char sequence and current accumulator value.
 */
public inline fun CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
public inline fun CharSequence.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
public inline fun CharSequence.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Splits this char sequence into a list of strings each not exceeding the given [size].
 * The last string in the resulting list may have less characters than the given [size].
 * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.
 */
@SinceKotlin("1.2")
public fun CharSequence.chunked(size: Int): List<String> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.
 * The last string in the resulting sequence may have less characters than the given [size].
 * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}

/**
 * Splits the original char sequence into pair of char sequences,
 * where *first* char sequence contains characters for which [predicate] yielded `true`,
 * while *second* char sequence contains characters for which [predicate] yielded `false`.
 */
@SinceKotlin("1.2")
public fun CharSequence.chunkedSequence(size: Int): Sequence<String> {
    return chunkedSequence(size) { it.toString() }
}

/**
 * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.
 * The last string in the resulting sequence may have less characters than the given [size].
 * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}
characters for which [predicate] yielded 'false':

```kotlin
@public inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first.toString(), second.toString())
}

/**
 * Splits the original string into two parts
 * containing characters for which [predicate] yielded 'true',
 * while *second* string contains characters for which [predicate] yielded 'false'.
 */
public inline fun String.partition(predicate: (Char) -> Boolean): Pair<String, String> {
    val first = StringBuilder()
    val second = StringBuilder()
    for (element in this) {
        if (predicate(element)) {
            first.append(element)
        } else {
            second.append(element)
        }
    }
    return Pair(first.toString(), second.toString())
}

/**
 * Returns a list of snapshots of the window of the given [size]
 * sliding along this char sequence with the given [step], where
 * each * snapshot is a string.
 */
@SinceKotlin("1.2")
public fun CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {
    return windowed(size, step, partialWindows) { it.toString() }
}

/**
 * Returns a list of results of applying the given
 * [transform] function to each char sequence representing a view over the window of the given [size]
 * sliding along this char sequence with the given [step].
 * Note that the char sequence passed to the [transform]
 * function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way,
 * unless you made a snapshot of it.
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {
    checkWindowSizeStep(size, step)
    val thisSize = this.length
    val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
    val result = ArrayList<R>(resultCapacity)
    var index = 0
    while (index in 0 until thisSize) {
        val end = index + size
        val coercedEnd = if (end < 0 || end > thisSize) { if (partialWindows) thisSize else break } else end
        result.add(transform(subSequence(index, coercedEnd)).toString())
        index += step
    }
    return result
}

/**
 * Returns a sequence of snapshots of the window of the given [size]
 * sliding along this char sequence with the given [step], where
 * each * snapshot is a string.
 */
@SinceKotlin("1.2")
public fun CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {
    return windowedSequence(size, step, partialWindows) { it.toString() }
}

/**
 * Returns a sequence of results of applying the given
 * [transform] function to each char sequence representing a view over the window of the given [size]
 * sliding along this char sequence with the given [step].
 * Note that the char sequence passed to the [transform]
 * function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way,
 * unless you made a snapshot of it.
 */
@SinceKotlin("1.2")
public fun CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false): Sequence<String> {
    return windowedSequence(size, step, partialWindows) { it.toString() }
}
```
elements to take in each window
* @param step the number of elements to move the window forward by on each
step, by default 1
* @param partialWindows controls whether or not to preserve partial windows
in the end if any.
* @param sample
samples.collections.Sequences.Transformations.averageWindows
* /n@SinceKotlin("1.2")/n
public fun <R>
CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false, transform:
(CharSequence) -> R): Sequence<R> { val windows = (if
(partialWindows) indices else 0 until length - size + 1) step step map { index ->
val end = index + size
val coercedEnd = if (end < 0 || end > length) length else end
transform(subSequence(index, coercedEnd)))
*/n
*/n
* Returns a list of pairs built from the characters of
* this and the [other] char sequences with the same index
* The returned list has length of the shortest char
sequence.*n*/n
*/n* @sample
samples.text.Strings.zipWithTransform
*/n
publication inline fun <V> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> V): List<V> {
    val size = length - 1
    if (size < 1) return emptyList()
    val result = ArrayList<V>(size)
    for (index in 0 until size) {
        result.add(transform(this[index], this[index + 1]))
    }
    return result
}

*/n* /n
*/n*/n* Returns a list containing the results of applying the given [transform] function
to each pair of two adjacent characters in this char sequence.*n*/n
*/n* The returned list is empty if this char sequence contains less than two
caracters.*n*/n
*/n* @sample
samples.collections.Collections.Transformations.zipWithNext
*/n
@SinceKotlin("1.2")/n
publication inline fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {
    val size = length - 1
    if (size < 1) return emptyList()
    val result = ArrayList<R>(size)
    for (index in 0 until size) {
        result.add(transform(this[index], this[index + 1]))
    }
    return result
}

*/n* /n
*/n*/n* Creates an [Iterable] instance that wraps the original char sequence returning its characters when being iterated.*n*/n
*/n*/n* /n
*/n*/n* Returns a sub sequence of this char sequence having
leading and trailing characters matching the [predicate] removed.*n*/n
*/n*/n* /n
*/n* Returns a string having leading and trailing characters matching the [predicate] removed.*n*/n
*/n*/n* /n
*/n*/n* Returns a string having leading and trailing characters matching the [predicate] removed.*n*/n
*/n*/n* /n
*/n* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.*n*/n*/n* Use of this source code is
governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file
*/n*/n* /n
*/n*/n* @file:kotlin.jvm.JvmMultifileClass*/n*/n* @file:kotlin.jvm.JvmName("StringsKt")*/n*/n* /n
*/n*/n* @package
kotlin.text*/n*/n*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* /n
*/n*/n* Open Source Used In DNAC Platform Arica 12153
fun String.trimStart(predicate: (Char) -> Boolean): String =
    (this as CharSequence).trimStart(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.
 */
public inline fun CharSequence.trimEnd(predicate: (Char) -> Boolean): String =
    (this as CharSequence).trimEnd(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading and trailing characters from the [chars] array removed.
 */
public fun String.trim(vararg chars: Char): String =
    trim { it in chars }

/**
 * Returns a sub sequence of this char sequence having leading characters from the [chars] array removed.
 */
public fun String.trim(vararg chars: Char): String =
    trimStart { it in chars }

/**
 * Returns a sub sequence of this char sequence having trailing characters from the [chars] array removed.
 */
public fun String.trim(vararg chars: Char): String =
    trimEnd { it in chars }

/**
 * Returns a sub sequence of this char sequence having leading whitespace removed.
 */
public fun String.trimStart(): String =
    trimStart(Char::isWhitespace)

/**
 * Returns a string having leading whitespace removed.
 */
public fun String.trim(): String =
    trim()
than zero.

```kotlin
if (length <= this.length) return this.subSequence(0, this.length)
val sb = StringBuilder(length)
    sb.append(this)
    for (i in 1..(length - this.length)) sb.append(padChar)
    return sb.toString()

/**
 * Pads the string to the specified [length] at the end with the specified character or space.
 * @param length the desired string length.
 * @param padChar [Char] to pad the character to pad string with, if it has length less than the [length] specified. Space is used by default.
 * @return a [String] a string of length at least [length] consisting of `this` string appended with [padChar] as many times as are necessary to reach that length.
 */
@kotlin.internal.InlineOnly
public fun String.padEnd(length: Int, padChar: Char): String = (this as CharSequence).padEnd(length, padChar).toString()

/**
 * Returns `true` if this nullable char sequence is either `null` or empty.
 * @return Returns `true` if this nullable char sequence is either `null` or empty.
 * @sample samples.text.Strings.padEnd
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

/**
 * Returns `true` if this char sequence is empty (contains no characters).
 * @return Returns `true` if this char sequence is empty (contains no characters).
 * @sample samples.text.Strings.isEmpty
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isEmpty(): Boolean = length == 0

/**
 * Returns `true` if this nullable char sequence is either `null` or empty or consists solely of whitespace characters.
 * @return Returns `true` if this nullable char sequence is either `null` or empty or consists solely of whitespace characters.
 * @sample samples.text.Strings.isNullOrBlank
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}
```

Open Source Used In DNAC Platform Arica 12155
warning\n@Deprecated("Use parameters named startIndex and endIndex.")\nReplaceWith("subSequence(startIndex = start, endIndex = end)")\npublic inline fun String.subSequence(start: Int, end: Int): CharSequence = subSequence(start, end)\n
@Deprecated("Use parameters named startIndex and endIndex.")\nReplaceWith("subSequence(startIndex = start, endIndex = end)\npublic inline fun String.subSequence(start: Int, end: Int): CharSequence = subSequence(start, end)\n
/**
 * Returns a substring of chars from a range of this char sequence starting at the
 * [startIndex] and ending right before the [endIndex].\n * @param startIndex the start index (inclusive).\n * @param endIndex the end index (exclusive). If not specified, the length of the char sequence is used.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.substring(startIndex: Int, endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()\n
/**
 * Returns a substring of chars at indices from the specified [range] of this char sequence.\n */\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.substring(range: IntRange): String = subSequence(range.start, range.endInclusive + 1).toString()\n
/**
 * Returns a substring before the first occurrence of [delimiter].\n */\npublic fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)\n}

/**
 * Returns a substring before the first occurrence of [delimiter].\n */\npublic fun String.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)\n}

/**
 * Returns a substring after the first occurrence of [delimiter].\n */\npublic fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)\n}

/**
 * Returns a substring after the first occurrence of [delimiter].\n */\npublic fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)\n}

/**
 * Returns a char sequence with content of this char sequence where its part at
 * the given range is replaced with the [replacement] char sequence.\n * @param startIndex the index of the first character to be replaced.\n * @param endIndex the index of the first character after the replacement to keep in the string.\n */\n@kotlin.internal.InlineOnly\npublic inline fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    if (endIndex < startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")\n    val sb = StringBuilder()
    sb.append(this, 0, startIndex)\n    sb.append(replacement)\n    sb.append(this, endIndex, length)\n    return sb.toString()\n}
replacement: CharSequence): String =
    (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()

/**
 * Returns a char sequence with content of this char sequence where its part at the
given [range] is replaced with the [replacement] char sequence.
 * The end index of the [range] is included in the part to be replaced.
 */
public fun CharSequence.replaceRange(range: IntRange, replacement: CharSequence): String =
    replaceRange(range.start, range.endInclusive + 1, replacement)

/**
 * Replace the part of string [range] with [replacement] string.
 * The end index of the [range] is included in the part to be replaced.
 */
@kotlin.internal.InlineOnly
public inline fun String.replaceRange(range: IntRange, replacement: String): String =
    (this as CharSequence).replaceRange(range, replacement).toString()
char sequence having this [delimiter] removed both from the start and end.
* Otherwise returns a new char sequence with the same characters.
public fun CharSequence.removeSurrounding(delimiter: CharSequence):
CharSequence = removeSurrounding(delimiter, delimiter)

public fun String.removeSurrounding(delimiter: CharSequence):
String = removeSurrounding(delimiter, delimiter)

public fun CharSequence.removeSurrounding(delimiter: CharSequence):
CharSequence = removeSurrounding(delimiter, delimiter)

/**
 * Removes the given [delimiter] string from both the start and the end of this string
 * if and only if it starts with and ends with the [delimiter].
 * Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(delimiter: CharSequence):
String = removeSurrounding(delimiter, delimiter)
sequence that matches the given regular expression\n * with the result of the given function \[transform\] that takes
[MatchResult] and returns a string to be used as a \n * replacement for that match.\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.replace(regex: Regex, noinline transform:
(MatchResult) -> CharSequence): String =\nregex.replace(this, transform)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.replaceFirst(regex: Regex, replacement:
String): String = regex.replaceFirst(this, replacement)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.matches(regex:
Regex): Boolean = regex.matches(this)\n
*\n@kotlin.internal.InlineOnly\ninternal fun CharSequence.regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean): Boolean {
    if ((otherOffset < 0) || (thisOffset < 0) || (thisOffset > this.length - length) ||
        (otherOffset > other.length - length)) {\n        return false\n    }\n    for (index in 0 until length) {\n        if (!this[thisOffset + index].equals(other[otherOffset + index], ignoreCase))\n            return false\n    }\n    return true\n}

*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.startsWith(char: Char, ignoreCase: Boolean = false): Boolean =\n    this.length > 0 &&
    this[0].equals(char, ignoreCase)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.endsWith(char: Char, ignoreCase: Boolean = false): Boolean =
    this.length > 0 &&
    this[lastIndex].equals(char, ignoreCase)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix)\n    else
        return regionMatchesImpl(0, prefix, 0, prefix.length, ignoreCase)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && suffix is String)
        return this.endsWith(suffix)\n    else
        return regionMatchesImpl(length - suffix.length, 0, 0, suffix.length, ignoreCase)\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val shortestLength = minOf(this.length, other.length)\n
    var i = 0\n    while (i < shortestLength && this[i].equals(other[i], ignoreCase)) {\n        // If this has Surrogate Pair At (i - 1) or other has Surrogate Pair At (i - 1)
        // \n    }\n    if (this.val shortestLength = minOf(this.length, otherLength)\n        val i = 0\n        while (i < shortestLength &&
            this[i].equals(other[i], ignoreCase) \n            { \n                // If this has Surrogate Pair At (i - 1) or other has Surrogate Pair At (i - 1)
                // \n            }\n            \n        return subSequence(thisLength - i, thisLength).toSequence()\n    }\n
*\n@kotlin.internal.InlineOnly\npublic inline fun CharSequence.commonSuffixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val shortestLength = minOf(this.length, otherLength)\n
    var i = 0\n    while (i < shortestLength &&
        this[i].equals(other[i], ignoreCase) \n        { \n            // If this has Surrogate Pair At (i - 1) or other has Surrogate Pair At (i - 1)
            // \n        }\n        \n    return subSequence(thisLength - i, thisLength).toSequence()\n}
optionally ignoring the case.\n
```kotlin
public fun findLastAnyOf(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int {\n    if (!ignoreCase && chars.size == 1 && this is String) {\n        val char = chars.single()\n        return nativeLastIndexOf(char, startIndex)\n    } else {\n        for (index in startIndex.coerceAtMost(lastIndex) downTo 0) {\n            if (other.regionMatches(0, this, index, other.length, ignoreCase))\n                return index\n        }\n    } else {\n        startIndex.coerceAtMost(lastIndex)\n    }\n}

private fun findAnyOf(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int {\n    if (!ignoreCase && strings.size == 1) {\n        val string = strings.single()\n        val index = if (!last) indexOf(string, startIndex) else -1\n        return index\n    } else {\n        for (index in startIndex.coerceAtLeast(0) downTo 0) {\n            if (other.regionMatches(0, this, index, other.length, ignoreCase))\n                return index\n        }\n    }
```

In the DNAC platform, Arica uses open-source libraries. These libraries help in manipulating strings, such as finding the index of a string within a sequence, ignoring character case, and starting the search from a specified index. This functionality is crucial for applications that require string manipulation, especially in a developer-friendly platform like DNAC.
starting from the specified [startIndex] and optionally ignoring the case.

@Param ignoreCase 'true' to ignore character case when matching a string. By default 'false'.

@Return An index of the first occurrence of matched string from [strings] or -1 if none of [strings] are found.

To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

the beginning to the end of this string, and finds at each position the first element in [strings] that matches this string at that position.

`*/*public fun CharSequence.indexOfAny(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int =

findAnyOf(strings, startIndex, ignoreCase, last = false).first ?: -1

* Returns the index within this string of the first occurrence of the specified character, starting from the specified

[startIndex].

@Param ignoreCase 'true' to ignore character case when matching a string. By default 'false'.

@Return An index of the last occurrence of matched string from [strings] or -1 if none of [strings] are found.

To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from

the end toward the beginning of this string, and finds at each position the first element in [strings] that matches this string at that position.

`*/*public fun CharSequence.lastIndexOfAny(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int =

lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase, last = true).first ?: -1

* Returns the index within this char sequence of the last occurrence of the specified character, starting from the specified

[startIndex].

@Param ignoreCase 'true' to ignore character case when matching a character. By default 'false'.

@Return An index of the first occurrence of [char] or -1 if none is found.

`*/*public fun CharSequence.indexOfAny(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {

if (ignoreCase || this !is String)

indexOf(charArrayOf(char), startIndex, ignoreCase)

else

nativeIndexOf(char, startIndex)

* Returns the index within this char sequence of the first occurrence of the specified

[string], at the specified [startIndex].

@Param ignoreCase 'true' to ignore character case when matching a string. By default 'false'.

@Return An index of the last occurrence of the specified character, starting from the specified

[startIndex].

@Param ignoreCase 'true' to ignore character case when matching a character. By default 'false'.

@Return An index of the first occurrence of [char] or -1 if none is found.

`*/*public fun CharSequence.lastIndexOfAny(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {

if (ignoreCase || this !is String)

lastIndexOf(charArrayOf(char), startIndex, ignoreCase)

else

nativeLastIndexOf(char, startIndex)

* Returns the index within this char sequence of the last occurrence of the specified

[string], at the specified [startIndex].

@Param ignoreCase 'true' to ignore character case when matching a character. By default 'false'.

@Return An index of the last occurrence of the specified character, starting from the specified

[startIndex].

`*/*public fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =

if (other is String)

indexOf(other, ignoreCase) >= 0

else

indexOf(other, 0, length, ignoreCase) >= 0

* Returns 'true' if this char sequence contains the specified [other] sequence of characters as a substring.

@Param ignoreCase 'true' to ignore character case when comparing strings. By default 'false'.

`*/*@Suppress("INAPPLICABLE_OPERATOR_MODIFIER")*/*public operator fun CharSequence.contains(char:
fun CharSequence.splitToSequence(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =
    rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 */
fun CharSequence.split(delimiter: Char, ignoreCase: Boolean = false, limit: Int = 0): List<String> {
    if (delimiter.toString().isBlank()) {
        return split(delimiter, ignoreCase, limit)
    }

    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).asIterable().map { substring(it) }
}

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiter].
 * This is specialized version of split which receives single non-empty delimiter and offers better performance.
 */
private fun CharSequence.split(delimiter: String, ignoreCase: Boolean, limit: Int): List<String> {
    // ...
**Open Source Used In DNAC Platform Arica**

String(chars: CharArray): String {
    var result = ""
    for (char in chars) {
        result += char
    }
    return result
}

/**
 * Converts the characters from a portion of the specified array to a string.
 *
 * @throws IndexOutOfBoundsException if either [offset] or [length] are less than zero or offset + length is out of [chars]
 * array bounds.
 */
@SinceKotlin("1.2")
@ExperimentalStdlibApi
public actual fun String(chars: CharArray, offset: Int, length: Int): String {
    String {
        if (offset < 0 || length < 0 || chars.size - offset < length)
            throw IndexOutOfBoundsException("size: ${chars.size}; offset: $offset; length: $length")
        var result = ""
        for (index in offset until offset + length) {
            result += chars[index]
        }
        return result
    }
}

/**
 * Concatenates characters in this [CharArray] into a String.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun CharArray.concatToString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}

/**
 * Concatenates characters in this [CharArray] or its subrange into a String.
 *
 * @param startIndex the beginning (inclusive) of the subrange of characters, 0 by default.
 * @param endIndex the end (exclusive) of the subrange of characters, size of this array by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun CharArray.concatToString(startIndex: Int = 0, endIndex: Int = this.size): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    var result = ""
    for (index in startIndex until endIndex) {
        result += this[index]
    }
    return result
}

/**
 * Returns a [CharArray] containing characters of this string.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}

/**
 * Returns a [CharArray] containing characters of this string or its substring.
 *
 * @param startIndex the beginning (inclusive) of the substring, 0 by default.
 * @param endIndex the end (exclusive) of the substring, length of this string by default.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.toCharArray(startIndex: Int = 0, endIndex: Int = this.length): CharArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return CharArray(endIndex - startIndex) { get(startIndex + it) }
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun ByteArray.decodeToString(): String {
    return decodeUtf8(this, 0, size, false)
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array or its subrange.
 *
 * @param startIndex the beginning (inclusive) of the subrange to decode, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to decode, length of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement char \uFFFD.
 *
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun ByteArray.decodeToString(
    startIndex: Int = 0,
    endIndex: Int = this.size,
    throwOnInvalidSequence: Boolean = false
): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return decodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(): ByteArray {
    return encodeUtf8(this, 0, length, false)
}

/**
 * Encodes this string or its substring to an array of bytes in UTF-8 encoding.
 *
 * @param startIndex the beginning (inclusive) of the substring to encode, 0 by default.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(startIndex: Int = 0, endIndex: Int = this.length): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}
@param endIndex the end (exclusive) of the substring to encode, length of this string by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char sequence or replace.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
 * @throws IllegalArgument Exception if [startIndex] is greater than [endIndex].
 * @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(
    startIndex: Int = 0,
    endIndex: Int = this.length,
    throwOnInvalidSequence: Boolean = false
): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return encodeUtf8(this, startIndex, endIndex,
        throwOnInvalidSequence)
}

/**
 * Returns a copy of this string converted to upper case using the rules of the default locale.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toUpperCase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to lower case using the rules of the default locale.
 */
@kotlin.internal.InlineOnly
public actual inline fun String.toLowerCase(): String = asDynamic().toLowerCase()

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeIndexOf(str: String, fromIndex: Int): Int = asDynamic().indexOf(str, fromIndex)

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeLastIndexOf(str: String, fromIndex: Int): Int = asDynamic().lastIndexOf(str, fromIndex)

@kotlin.internal.InlineOnly
internal inline fun String.nativeStartsWith(s: String, position: Int): Boolean = asDynamic().startsWith(s, position)

@kotlin.internal.InlineOnly
internal inline fun String.nativeEndsWith(s: String): Boolean = asDynamic().endsWith(s)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int): String = asDynamic().substring(startIndex)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int, endIndex: Int): String = asDynamic().substring(startIndex, endIndex)

@kotlin.internal.InlineOnly
public inline fun String.concat(str: String): String = asDynamic().concat(str)

@kotlin.internal.InlineOnly
public inline fun String.match(regex: String): Array<String>? = asDynamic().match(regex)

private val STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b -> a.compareTo(b, ignoreCase = true) }

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String>
    get() = STRING_CASE_INSENSITIVE_ORDER

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UByteArray
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@PublishedApi
internal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> {

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    internal constructor(size: Int) : this(ByteArray(size))

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     */
    operator fun get(index: Int): UByte = get0(index)

    fun get0(index: Int) = asUByteArray().get0(index)

    operator fun set(index: Int, value: UByte) = asUByteArray().set0(index, value)

    fun set0(index: Int, value: UByte) = asUByteArray().set0(index, value)

    operator fun plus(other: UByteArray): UByteArray = asUByteArray().plus(other)

    fun plus(other: UByteArray): UByteArray = asUByteArray().plus(other)

    operator fun minus(other: UByteArray): UByteArray = asUByteArray().minus(other)

    fun minus(other: UByteArray): UByteArray = asUByteArray().minus(other)

    operator fun times(factor: Int): UByteArray = asUByteArray().times(factor)

    fun times(factor: Int): UByteArray = asUByteArray().times(factor)

    operator fun times(factor: UByte): UByteArray = asUByteArray().times(factor)

    fun times(factor: UByte): UByteArray = asUByteArray().times(factor)

    override fun equals(other: Any?): Boolean = (this == other)

    override fun hashCode(): Int = hashcode()

    override fun hashcode(): Int = hashcode()

    override fun toString(): String = asUByteArray().toString()
[IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
 * public operator fun get(index: Int): UByte = storage[index].toUByte()
 /**
 * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
 */
 public operator fun set(index: Int, value: UByte) {
     storage[index] = value.toByte()
 }
/**
 * Returns the number of elements in the array.
 */
 public override val size: Int get() = storage.size
 /**
 * Creates an iterator over the elements of the array.
 */
 public override operator fun iterator(): iterator() = Iterator(storage)
 private class Iterator(private var array: ByteArray) : UByteIterator() {
     override fun hasNext() = index < array.size
     override fun nextUByte() = if (index < array.size)
         array[index++].toUByte() else throw NoSuchElementException(index.toString())
 }
 /**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 */
 fun ByteArray(size: Int, init: (Int) -> UByte): ByteArray
 }

fun ubyteArrayOf(vararg elements: UByte): UByteArray

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun

should return the value for an array element given its index.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
    return UIntArray(IntArray(size) { index ->
        init(index).toInt()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun uintArrayOf(vararg elements: UInt): UIntArray = elements
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
import kotlin.experimental.*

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@PublishedApi internal constructor(@PublishedApi internal val storage: LongArray): Collection<ULong> {
    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) :
        this(LongArray(size))

    /**
     * Returns the array element at the given [index]. This method can be called using
     * the index operator. *
     */
    public operator fun get(index: Int): ULong = storage[index].toULong()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called
     * using the index operator. *
     */
    public operator fun set(index: Int, value: ULong) {
        storage[index] = value.toLong()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): ULongIterator =
        Iterator(storage)

    private class Iterator(private val array: LongArray) : ULongIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextULong() = if (index < array.size)
            array[index++].toULong() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: ULong): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        // If (element as Any?) !is ULong) return false
        return storage.contains(element.toLong())
    }

    override fun containsAll(elements: Collection<ULong>): Boolean {
        return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function. [init] should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}
```

```kotlin
kotlin.experimental.*

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}
```

```kotlin
kotlin.experimental.*

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}
```

```kotlin
kotlin.experimental.*

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}
```

```kotlin
kotlin.experimental.*

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements
```
form.

```kotlin
public const val SIZE_BITS: Int = 64

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
 * it's less than other, or a positive number if it's greater than other.
 **/
public const val SIZE_BITS: Int = 64
```

```kotlin
/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
 * it's less than other, or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toULong())

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
 * it's less than other, or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toULong())

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
 * it's less than other, or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toULong())

/**
 * Compares this value with the specified value for order.
 * Returns zero if this value is equal to the specified other value, a negative number if
 * it's less than other, or a positive number if it's greater than other.
 */
@kotlin.internal.InlineOnly
public inline operator fun compareTo(other: ULong): Int = this.compareTo(other.toULong())

/**
 * Adds the other value to this value. */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())

/**
 * Adds the other value to this value. */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())

/**
 * Adds the other value to this value. */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())

/**
 * Adds the other value to this value. */
@kotlin.internal.InlineOnly
public inline operator fun plus(other: ULong): ULong = this.plus(other.toULong())

/**
 * Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())

/**
 * Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())

/**
 * Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())

/**
 * Subtracts the other value from this value. */
@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong = this.minus(other.toULong())

/**
 * Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): ULong = this.times(other.toULong())

/**
 * Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): ULong = this.times(other.toULong())

/**
 * Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): ULong = this.times(other.toULong())

/**
 * Multiplies this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong = this.times(other.toULong())

/**
 * Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): ULong = this.div(other.toULong())

/**
 * Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): ULong = this.div(other.toULong())

/**
 * Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): ULong = this.div(other.toULong())

/**
 * Divides this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong = this.div(other.toULong())

/**
 * Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())

/**
 * Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())

/**
 * Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())

/**
 * Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.rem(other.toULong())
```
public inline operator fun dec(): ULong = ULong(data.dec())

/** Creates a range from this value to the specified [other] value. */
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)

/** Shifts this value left by the [bitCount] number of bits. */
@kotlin.internal.InlineOnly
public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)

/** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)

/** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)

/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)

/** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)

/** Converts this [ULong] value to [Byte]. */
@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [ULong] value to [UByte]. */
@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [ULong] value to [UShort]. */
@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [ULong] value to [UInt]. */
@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [ULong] value to [Int]. */
@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Converts this [ULong] value to [Long]. */
@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()

/** Converts this [ULong] value to [Short]. */
@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Converts this [ULong] value to [Byte.MAX_VALUE], the resulting `Byte` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toByteMaxValue(): Byte = Byte.MAX_VALUE

/** Converts this [ULong] value to [UByte.MAX_VALUE], the resulting `UByte` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toUByteMaxValue(): UByte = UByte.MAX_VALUE

/** Converts this [ULong] value to [UShort.MAX_VALUE], the resulting `UShort` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toUShortMaxValue(): UShort = UShort.MAX_VALUE

/** Converts this [ULong] value to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toUIntMaxValue(): UInt = UInt.MAX_VALUE

/** Converts this [ULong] value to [Int.MAX_VALUE], the resulting `Int` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toIntMaxValue(): Int = Int.MAX_VALUE

/** Converts this [ULong] value to [Long.MAX_VALUE], the resulting `Long` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toLongMaxValue(): Long = Long.MAX_VALUE

/** Converts this [ULong] value to [Short.MAX_VALUE], the resulting `Short` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toShortMaxValue(): Short = Short.MAX_VALUE

/** Converts this [ULong] value to [Byte().toString], the resulting `Byte` value represents the same numerical value as this `ULong`. */
@kotlin.internal.InlineOnly
public inline fun toByteString(): String = data.toString()
Converts this [Byte] value to [ULong].
* If this value is positive, the resulting `ULong` value represents the same numerical value as this `Byte` value.
* The least significant 8 bits of the resulting `ULong` value are the same as the bits of this `Byte` value, whereas the most significant 56 bits are filled with the sign bit of this value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toULong(): ULong = ULong(this.toLong())
```

Converts this [Short] value to [ULong].
* If this value is positive, the resulting `ULong` value represents the same numerical value as this `Short` value.
* The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `Int` value, whereas the most significant 32 bits are filled with the sign bit of this value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toULong(): ULong = ULong(this.toLong())
```

Converts this [Int] value to [ULong].
* The least significant 32 bits of the resulting `ULong` value are the same as the bits of this `Int` value.
* The least significant 32 bits are filled with the sign bit of this value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toULong(): ULong = ULong(this.toLong())
```

Converts this [Long] value to [ULong].
* If this value is positive, the resulting `ULong` value represents the same numerical value as this `Long` value.
* The resulting `ULong` value has the same binary representation as this `Long` value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toULong(): ULong = ULong(this)
```

Converts this [Float] value to [ULong].
* The fractional part, if any, is rounded down towards zero.
* Returns zero if this `Float` value is negative or `NaN`.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())
```

Converts this [Double] value to [ULong].
* The fractional part, if any, is rounded down towards zero.
* Returns zero if this `Double` value is negative or `NaN`, otherwise.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toULong(): ULong = doubleToULong(this)
```

Open Source Used In DNAC Platform Arica 12170
Collection<UnsignedShort>): Boolean {
    return (elements as Collection<*>).all { it is UnsignedShort &&
        storage.contains(it.toShort()) }
}

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function. The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index ->
        init(index).toShort()
    })
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function. The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UnsignedShort): UShortArray = elements

UShortArray.component4(): UShort {\n    return get(3)\n}\n/**\n * Returns 5th *element* from the collection.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UIntArray.component5(): UInt {\n    return get(4)\n}\n/**\n * Returns 5th *element* from the collection.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun ULongArray.component5(): ULong {\n    return get(4)\n}\n/**\n * Returns 5th *element* from the collection.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UByteArray.component5(): UByte {\n    return get(4)\n}\n/**\n * Returns 5th *element* from the collection.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline operator fun UShortArray.component5(): UShort {\n    return get(4)\n}\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UIntArray.elementAt(index: Int): UInt

/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun ULongArray.elementAt(index: Int): ULong

/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UByteArray.elementAt(index: Int): UByte

/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic expect fun UShortArray.elementAt(index: Int): UShort

/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**\n * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.elementAtOrNull(index: Int): UInt? {
    return this.getOrNull(index)
}

/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.elementAtOrNull(index: Int): ULong? {
    return this.getOrNull(index)
}

/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.elementAtOrNull(index: Int): UByte? {
    return this.getOrNull(index)
}

/**\n * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.elementAtOrNull(index: Int): UShort? {
    return this.getOrNull(index)
}

Open Source Used In DNAC Platform Arica 12172
/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.elementAtOrNull(index: Int): ULong? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.elementAtOrNull(index: Int): UByte? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.elementAtOrNull(index: Int): UShort? {
    return this.getOrNull(index)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.first(): UInt {
    return storage.first().toUInt()
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.first(): ULong {
    return storage.first().toULong()
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.first(): UByte {
    return storage.first().toUByte()
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.first(): UShort {
    return storage.first().toUShort()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.firstOrNull(): UInt? {
    return storage.firstOrNull().toUInt()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.firstOrNull(): ULong? {
    return storage.firstOrNull().toULong()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.firstOrNull(): UByte? {
    return storage.firstOrNull().toUByte()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 * 
 * @sample samples.collections.Collections.Elements.first()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.firstOrNull(): UShort? {
    return storage.firstOrNull().toUShort()
}

/**
 * Returns last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {
    return lastOrNull(predicate)
}

/**
 * Returns last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.findLast(predicate: (ULong) -> Boolean): ULong? {
    return lastOrNull(predicate)
}

/**
 * Returns last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.findLast(predicate: (UByte) -> Boolean): UByte? {
    return lastOrNull(predicate)
}

/**
 * Returns last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.findLast()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.last(): UInt {
    return storage.last().toUInt()
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.last(): ULong {
    return storage.last().toULong()
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.last(): UByte {
    return storage.last().toUByte()
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 * 
 * @sample samples.collections.Collections.Elements.last()
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.last(): UShort {
    return storage.last().toUShort()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.firstOrNull(): UInt? {
    return storage.firstOrNull().toUInt()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.firstOrNull(): ULong? {
    return storage.firstOrNull().toULong()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.firstOrNull(): UByte? {
    return storage.firstOrNull().toUByte()
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.firstOrNull(): UShort? {
    return storage.firstOrNull().toUShort()
}
UIntArray.first(predicate: (UInt) -> Boolean): UInt {  
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}  

>Returns the first element matching the given [predicate], or `null` if no such element is found.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

ULongArray.first(predicate: (ULong) -> Boolean): ULong {  
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}  

>Returns the first element matching the given [predicate], or `null` if no such element is found.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UByteArray.first(predicate: (UByte) -> Boolean): UByte {  
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}  

>Returns the first element matching the given [predicate], or `null` if no such element is found.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UShortArray.first(predicate: (UShort) -> Boolean): UShort {  
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}  

>Returns the first element matching the given [predicate], or `null` if no such element is found.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UIntArray.firstOrNull(): UInt? {
    return if (isEmpty()) null else this[0]
}  

>Returns the first element, or `null` if the array is empty.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

ULongArray.firstOrNull(): ULong? {
    return if (isEmpty()) null else this[0]
}  

>Returns the first element, or `null` if the array is empty.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UByteArray.firstOrNull(): UByte? {
    return if (isEmpty()) null else this[0]
}  

>Returns the first element, or `null` if the array is empty.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UShortArray.firstOrNull(): UShort? {
    return if (isEmpty()) null else this[0]
}  

>Returns the first element, or `null` if the array is empty.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}  

>Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}  

>Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}  

>Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}  

>Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun

UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}  

(Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
lastIndex) get(index) else defaultValue(index)/n}*/n

*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\nv return if (index >= 0 && & index <= lastIndex) get(index) else defaultValue(index)/n}*/n

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\nv return if (index >= 0 && & index <= lastIndex) get(index) else defaultValue(index)/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.getOrNull(index: Int): Uint? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.getOrNull(index: Int): ULong? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline fun
UByteArray.getOrNull(index: Int): UByte? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline fun
UShortArray.getOrNull(index: Int): UShort? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.indexOf(element: Uint): Int {\nv return storage.indexOf(element.toInt()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.indexOf(element: ULong): Int {\nv return storage.indexOf(element.toLong()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.indexOf(element: UByte): Int {\nv return storage.indexOf(element.toByte()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.indexOf(element: UShort): Int {\nv return storage.indexOf(element.toShort()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.indexOfFirst(predicate: (Uint) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toInt()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toULong()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toUByte()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toUShort()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toUByte()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toUShort/)n}*/n

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {\nv return if (index >= 0 && & index <= lastIndex) get(index) else defaultValue(index)/n}*/n

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun
UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {\nv return if (index >= 0 && & index <= lastIndex) get(index) else defaultValue(index)/n}*/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.getOrNull(index: Int): Uint? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.getOrNull(index: Int): ULong? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.getOrNull(index: Int): UByte? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.getOrNull(index: Int): UShort? {\nv return if (index >= 0 && & index <= lastIndex) get(index) else null}/n

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.indexOf(element: Uint): Int {\nv return storage.indexOf(element.toInt()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.indexOf(element: ULong): Int {\nv return storage.indexOf(element.toLong()/n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.indexOf(element: UByte): Int {\nv return storage.indexOf(element.toByte/)n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.indexOf(element: UShort): Int {\nv return storage.indexOf(element.toShort/)n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain such element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UintArray.indexOfFirst(predicate: (Uint) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toInt()) \n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain such element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic function ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {\nv return storage.indexOfFirst { predicate(it.toULong()) \n}*/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain such element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic function UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {\nv return 0}/n

/**
 * Returns an element at the given [index] or `null` if the array does not contain such element.
 */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic function UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {\nv return 0}/n
/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int {
    return storage.indexOfLast { predicate(it.toUInt()) }
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {
    return storage.indexOfLast { predicate(it.toULong()) }
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {
    return storage.indexOfLast { predicate(it.toUByte()) }
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {
    return storage.indexOfLast { predicate(it.toUShort()) }
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.last(): UInt {
    return storage.last().toUInt()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongArray.last(): ULong {
    return storage.last().toULong()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UByteArray.last(): UByte {
    return storage.last().toUByte()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.last(): UShort {
    return storage.last().toUShort()
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf(element.toInt())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf { predicate(it.toUInt()) }
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongArray.lastIndexOf(element: ULong): Int {
    return storage.lastIndexOf { predicate(it.toULong()) }
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf { predicate(it.toUByte()) }
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.lastIndexOf(element: UShort): Int {
    return storage.lastIndexOf { predicate(it.toUShort()) }
}
last index of \[element\], or \(-1\) if the array does not contain element.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun ULongArray.lastIndexOf(element: ULong): Int { \n    return storage.lastIndexOf(element.toLong())\n}

@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UByteArray.lastIndexOf(element: UByte): Int { \n    return storage.lastIndexOf(element.toByte())\n}

@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UShortArray.lastIndexOf(element: UShort): Int { \n    return storage.lastIndexOf(element.toShort())\n}
```

Returns last index of \[element\], or \(-1\) if the array does not contain element.

```kotlin
/**
 * Returns last index of \[element\], or \(-1\) if the array does not contain element.
 */
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UIntArray.lastIndexOf(element: UInt): Int { \n    return storage.lastIndexOf(element)\n}
```

Returns last element, or \(null\) if the array is empty.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@null fun UIntArray.lastOrNull(): UInt? { \n    return if (isEmpty()) null else this[size - 1]\n}
```

Returns the last element, or \(null\) if the array is empty.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@null fun ULongArray.lastOrNull(): ULong? { \n    return if (isEmpty()) null else this[size - 1]\n}
```

Returns the last element, or \(null\) if the array is empty.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@null fun UByteArray.lastOrNull(): UByte? { \n    return if (isEmpty()) null else this[size - 1]\n}
```

Returns the last element, or \(null\) if no such element was found.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@null fun UShortArray.lastOrNull(): UShort? { \n    return if (isEmpty()) null else this[size - 1]\n}
```

Returns the last element matching the given \[predicate\], or \(null\) if no such element was found.

```kotlin
/**
 * Returns the last element matching the given \[predicate\], or \(null\) if no such element was found.
 */
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? {
    for (index in this.indices.reversed()) { \n        val element = this[index]\n        if (predicate(element)) return element\n    }
    return null
}
```

Returns the last element matching the given \[predicate\], or \(null\) if no such element was found.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {
    for (index in this.indices.reversed()) { \n        val element = this[index]\n        if (predicate(element)) return element\n    }
    return null
}
```

Returns the last element matching the given \[predicate\], or \(null\) if no such element was found.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (index in this.indices.reversed()) { \n        val element = this[index]\n        if (predicate(element)) return element\n    }
    return null
}
```

Returns the last element matching the given \[predicate\], or \(null\) if no such element was found.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (index in this.indices.reversed()) { \n        val element = this[index]\n        if (predicate(element)) return element\n    }
    return null
}
```

Returns a random element from this array.

```kotlin
/**
 * Returns a random element from this array.
 * \n * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UIntArray.random(): UInt
```

Returns a random element from this array.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun ULongArray.random(): ULong
```

Returns a random element from this array.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UByteArray.random(): UByte
```

Returns a random element from this array.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@kotlin.internal.InlineOnly\npublic inline fun UShortArray.random(): UShort
```

Returns a random element from this array using the specified source of randomness.

```kotlin
@SinceKotlin("1.3.3")@ExperimentalUnsignedTypes@null fun UIntArray.random(random: Random): UInt
```

Returns a random element from this array.
### Open Source Used in DNAC Platform Arica

```kotlin
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return
```

```kotlin
/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.random(random: Random): ULong {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return
```

```kotlin
/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.random(random: Random): UByte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return
```

```kotlin
/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.random(random: Random): UShort {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return
```

```kotlin
/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(): UInt {
    return storage.single().toUInt()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(): ULong {
    return storage.single().toULong()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.single(): UByte {
    return storage.single().toUByte()
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.single(): UShort {
    return storage.single().toUShort()
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as UInt
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.single(predicate: (ULong) -> Boolean): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as ULong
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as UByte
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UShort
}
@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {\n    var single: UShort? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException("Array contains more than one matching element.\")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.\")\n    return single \n}

/**\n * Returns single element, or ‘null’ if the array is empty or has more than one element.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UIntArray.singleOrNull(): UInt? {\n    return if (size == 1) this[0] else null\n}

/**\n * Returns single element, or ‘null’ if the array is empty or has more than one element.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun ULongArray.singleOrNull(): ULong? {\n    return if (size == 1) this[0] else null\n}

/**\n * Returns single element, or ‘null’ if the array is empty or has more than one element.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UByteArray.singleOrNull(): UByte? {\n    return if (size == 1) this[0] else null\n}

/**\n * Returns single element, or ‘null’ if the array is empty or has more than one element.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UShortArray.singleOrNull(): UShort? {\n    return if (size == 1) this[0] else null\n}

/**\n * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\nfun UIntArray.singleOrNull(predicate: (UInt) -> Boolean): UInt? {\n    var single: UInt? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single
}

/**\n * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\nfun ULongArray.singleOrNull(predicate: (ULong) -> Boolean): ULong? {\n    var single: ULong? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single
}

/**\n * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\nfun UByteArray.singleOrNull(predicate: (UByte) -> Boolean): UByte? {\n    var single: UByte? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single
}

/**\n * Returns the single element matching the given [predicate], or ‘null’ if element was not found or more than one element was found.\n */\n@.SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\nfun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {\n    var single: UShort? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) return null\n            single = element\n            found = true\n        }\n    }\n    if (!found) return null\n    return single
}

/**\n * Returns a list containing all elements except first [n] elements.\n */\n@throws IllegalArgumentException if [n] is negative.\n
* @sample samples.collections.Collections.Transformations.drop

* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UIntArray.drop(n: Int): List<UInt> {\n    require(n >= 0) { "Requested element count $n is less than zero.\" \n} \n    return takeLast((size - n).coerceAtLeast(0))\n}

* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun ULongArray.drop(n: Int): List<ULong> {\n    require(n >= 0) { "Requested element count $n is less than zero.\" \n} \n    return takeLast((size - n).coerceAtLeast(0))\n}

* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UByteArray.drop(n: Int): List<UByte> {\n    require(n >= 0) { "Requested element count $n is less than zero.\" \n} \n    return takeLast((size - n).coerceAtLeast(0))\n}

* @SinceKotlin("1.3")@ExperimentalUnsignedTypes\nfun UShortArray.drop(n: Int): List<UShort> {\n    require(n >= 0) { "Requested element count $n is less than zero.\" \n} \n    return takeLast((size - n).coerceAtLeast(0))\n}

Open Source Used In DNAC Platform Arica 12179
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.drop(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." } return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.drop(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." } return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.dropLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." } return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.dropLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." } return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.dropLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last [n] elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.dropLastWhile(predicate: (ULong) -> Boolean): List<ULong> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last [n] elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last [n] elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.dropLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
public inline fun UIntArray.dropWhile(predicate: (UInt) -> Boolean): List<UInt> {
    var yielding = false
    val list = ArrayList<UInt>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {
    var yielding = false
    val list = ArrayList<ULong>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropWhile(predicate: (UByte) -> Boolean): List<UByte> {
    var yielding = false
    val list = ArrayList<UByte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.dropWhile(predicate: (UShort) -> Boolean): List<UShort> {
    var yielding = false
    val list = ArrayList<UShort>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {
    return filterTo(ArrayList<UInt>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {
    return filterTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {
    return filterTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {
    return filterTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {
    return filterIndexedTo(ArrayList<UInt>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {
    return filterIndexedTo(ArrayList<UInt>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(), index, predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), index, predicate)
}
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {
    return filterIndexedTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {
    return filterIndexedTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterNot(predicate: (UInt) -> Boolean): List<UInt> {
    return filterNotTo(ArrayList<UInt>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterNot(predicate: (ULong) -> Boolean): List<ULong> {
    return filterNotTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterNot(predicate: (UByte) -> Boolean): List<UByte> {
    return filterNotTo(ArrayList<UByte>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UShortArray.filterNot(predicate: (UShort) -> Boolean): List<UShort> {
    return filterNotTo(ArrayList<UShort>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UIntArray.filterIndexedTo(destination: MutableList<in UInt>?, predicate: (index: Int, UInt) -> Boolean): MutableList<in UInt> {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination?.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexedTo(destination: MutableList<in ULong>?, predicate: (index: Int, ULong) -> Boolean): MutableList<in ULong> {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination?.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun UByteArray.filterIndexedTo(destination: MutableList<in UByte>?, predicate: (index: Int, UByte) -> Boolean): MutableList<in UByte> {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination?.add(element)
    }
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination?.add(element)
    }
    return destination
}
(element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByteArray>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (UInt) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UByteArray>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.slice(indices: IntRange): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.slice(indices: IntRange): List<ULong> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: IntRange): List<UByte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.slice(indices: Iterable<Int>): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.slice(indices: Iterable<Int>): List<UShort> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.slice(indices: Iterable<Int>): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.slice(indices: Iterable<Int>): List<ULong> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.slice(indices: Iterable<Int>): List<UShort> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.slice(indices: Iterable<Int>): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
require(n >= 0) { "Requested element count $n is less than zero." }
if (n == 0) return emptyList()
if (n >= size) return toList()
if (n == 1) return listOf(this[0])
for (index in size - n until size) list.add(this[index])
return list

/**
 * Returns a list containing last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.takeLast
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    val size = size
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    return ArrayList(n)
    for (index in size - n until size) list.add(this[index])
    return list
}

/**
 * Returns a list containing last elements satisfying the given [predicate].
 * 
 * @throws IllegalArgumentException if [predicate] is null.
 * 
 * @sample samples.collections.Collections.Transformations.takeLast
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    return ArrayList-lastWhile(predicate)
}

@Sample
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    return ArrayList-
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

open Source Used In DNAC Platform Arica 12185
Returns a list containing last elements satisfying the given [predicate].

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    return toList() // Returns a list containing first elements satisfying the given [predicate].
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    return toList() // Returns a list containing first elements satisfying the given [predicate].
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reverse(): Unit {
    storage.reverse()
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reverse(): Unit {
    storage.reverse()
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reverse(): Unit {
    storage.reverse()
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reversed(): List<UInt> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reversed(): List<ULong> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reversed(): List<UByte> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reversed(): List<UShort> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reverseArray(): UIntArray {
    return UIntArray(storage.reversedArray())
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reverseArray(): UIntArray {
    return UIntArray(storage.reversedArray())
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reverseArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}
```

```kotlin
*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reverseArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}
```
ULongArray.reversedArray(): ULongArray {
    return ULongArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UByteArray.reversedArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UShortArray.reversedArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UIntArray.reversedArray(): UIntArray {
    return UIntArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UByteArray.reversedArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UShortArray.reversedArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun

UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun ULongArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort() reverse()
    }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UIntArray.sorted(): List<UInt> {
    return copyOf().apply { sort() }.asList()
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun ULongArray.sorted(): List<ULong> {
    return copyOf().apply { sort() }.asList()
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UByteArray.sorted(): List<UByte> {
    return copyOf().apply { sort() }.asList()
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UShortArray.sorted(): List<UShort> {
    return copyOf().apply { sort() }.asList()
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun ULongArray.sortedArray(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UByteArray.sortedArray(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

*An@SinceKotlin("1.3")@ExperimentalUnsignedTypes\npublic fun UShortArray.sortedArray(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UShortArray.sortedArrayDescending():
UShortArray { if (isEmpty()) return this return copyOf().apply { sortDescending() } }

Returns a list of all elements sorted descending according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UIntArray.sortedDescending(): List<UInt>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun ULongArray.sortedDescending(): List<ULong>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UByteArray.sortedDescending(): List<UByte>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UShortArray.sortedDescending(): List<UShort>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UIntArray.sortedDescending(): List<UInt>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun ULongArray.sortedDescending(): List<ULong>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UByteArray.sortedDescending(): List<UByte>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UShortArray.sortedDescending(): List<UShort>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UIntArray.sortedDescending(): List<UInt>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun ULongArray.sortedDescending(): List<ULong>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UByteArray.sortedDescending(): List<UByte>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UShortArray.sortedDescending(): List<UShort>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UIntArray.sortedDescending(): List<UInt>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun ULongArray.sortedDescending(): List<ULong>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UByteArray.sortedDescending(): List<UByte>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun UShortArray.sortedDescending(): List<UShort>
{ return copyOf().apply { sort() }.reversed() }

Returns a list of all elements sorted according to their natural sort order.
[UShortArray], which is a view of this array where each element is an unsigned reinterpretation
* of the corresponding element of this array.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ShortArray.asUShortArray(): UShortArray {
    return UShortArray(this)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another;\n * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UIntArray.contentEquals(other: UIntArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another;\n * i.e. contain the same number of the same elements in the same order.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongArray.contentEquals(other: ULongArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns a hash code based on the contents of this array as if it is \[List\].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is \[List\].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is \[List\].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is \[List\].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.contentToString(): String {
    return joinToString(", ", "[", "]")
}

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.contentToString(): String {
    return joinToString(", ", "[", "]")
}

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.contentToString(): String {
    return joinToString(", ", "[", "]")
}

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.contentToString(): String {
    return joinToString(", ", "[", "]")
}
Open Source Used In DNAC Platform Arica

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UIntArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
ULongArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UByteArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of
 * range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size):
UShortArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.copyOf(): UIntArray {
    return UIntArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.copyOf(): ULongArray {
    return ULongArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.copyOf(): UByteArray {
    return UByteArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.copyOf(): UShortArray {
    return UShortArray(storage.copyOf())
}
```

12190
UByteArray.copyOf(): UByteArray {
    return UByteArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array.
 * 
 * @param sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOf(): UByteArray {
    return UByteArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOf(newSize: Int): UByteArray {
    return UByteArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOf(): UShortArray {
    return UShortArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOf(newSize: Int): UShortArray {
    return UShortArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Returns new array which is a copy of the specified range of the original array.
 * 
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray {
    return UShortArray(storage.copyOfRange(fromIndex, toIndex))
}
NonNull copyOfRange(fromIndex, toIndex))

/**
* Fills this array or its subrange with the specified [element] value.
* @param fromIndex the start of the range (inclusive), 0 by default.
* @param toIndex the end of the range (exclusive), size of this array by default.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws IllegalArgumentOutOfRangeException if [fromIndex] is greater than [toIndex].
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toShort(), fromIndex, toIndex)
}

/**
* Fills this array or its subrange with the specified [element] value.
* @param fromIndex the start of the range (inclusive), 0 by default.
* @param toIndex the end of the range (exclusive), size of this array by default.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws IllegalArgumentOutOfRangeException if [fromIndex] is greater than [toIndex].
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toByte(), fromIndex, toIndex)
}

/**
* Fills this array or its subrange with the specified [element] value.
* @param fromIndex the start of the range (inclusive), 0 by default.
* @param toIndex the end of the range (exclusive), size of this array by default.
* @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
* @throws IllegalArgumentOutOfRangeException if [fromIndex] is greater than [toIndex].
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toLong(), fromIndex, toIndex)
}

/**
* Returns the range of valid indices for the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UIntArray.indices: IntRange
    get() = storage.indices

/**
* Returns the range of valid indices for the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val ULongArray.indices: IntRange
    get() = storage.indices

/**
* Returns the range of valid indices for the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UByteArray.indices: IntRange
    get() = storage.indices

/**
* Returns the range of valid indices for the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.indices: IntRange
    get() = storage.indices

/**
* Returns an array containing all elements of the original array and then the given [element].
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.plus(element: UInt): UIntArray {
    return UIntArray(storage + element.toInt())
}

/**
* Returns an array containing all elements of the original array and then the given [element].
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage + element.toLong())
}
public inline operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage + element.toByte())
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return UIntArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return ULongArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
}

public fun UIntArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

public fun ULongArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

public fun UByteArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

public fun UShortArray.sort(): Unit {
    if (size > 1) sortArray(this)
}
ByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [IntArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.toIntArray(): IntArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.toULongArray(): ULongArray {
    return ULongArray(size) { index -> this[index] }
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Array<out UByte>.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UByteArray], which is a copy of this array where each element is an unsigned
 * reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.toLongArray(): LongArray {
    return storage.copyOf()
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.toUIntArray(): UIntArray {
    return UIntArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UIntArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun IntArray.toUIntArray(): UIntArray {
    return UIntArray(this.copyOf())
}

/**
 * Returns an array of ULong containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Array<out UByte>.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.toULongArray(): ULongArray {
    return ULongArray(size) { index -> this[index] }
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Array<out UByte>.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UByteArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.toLongArray(): LongArray {
    return storage.copyOf()
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
}

/**
 * Returns an array of type
 * [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.toUIntArray(): UIntArray {
    return UIntArray(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UIntArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun IntArray.toUIntArray(): UIntArray {
    return UIntArray(this.copyOf())
}

/**
 * Returns an array of ULong containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type
 * [ULongArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns an array of UByte containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Array<out UByte>.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UintArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns an array of type
 * [UIntArray], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun IntArray.toUIntArray(): UIntArray {
    return UIntArray(this.copyOf())
}

/**
 * Returns an array of ULong containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public func
ShortArray.toUShortArray(): UShortArray

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R>

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R>

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R>

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R> UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R>

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : MutableCollection<in R>> UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable<R>): C

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <K> UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable<R>): C

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <K> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <K> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <K> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C

Open Source Used In DNAC Platform Arica 12195
groupByTo(LinkedHashMap<K, MutableList<ULong>>(), keySelector)\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupBy\n
*/
@SinceKotlin("1.3")\npublic inline fun <K> UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> \n
groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupBy\n
*/
@SinceKotlin("1.3")\npublic inline fun <K> UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> \n
groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)\n
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\nThe returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n
*/
@SinceKotlin("1.3")\npublic inline fun <K, V> UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> \n
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and returns a map where each group key is associated with a list of corresponding values.\nThe returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupByKeysAndValues\n
*/
@SinceKotlin("1.3")\npublic inline fun <K, V> ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> \n
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\nThe returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupBy\n
*/
@SinceKotlin("1.3")\npublic inline fun <K> UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> \n
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\nThe returned map preserves the entry iteration order of the keys produced from the original array.\n * @sample samples.collections.Collections.Transformations.groupBy\n
*/
@SinceKotlin("1.3")\npublic inline fun <K> UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>> \n
groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\nThe returned map preserves the entry iteration order of the keys produced from the original array.\n * @return The
[destination] map.

```kotlin
@Sample
val key = keySelector(element)
val list = destination.getOrPut(key)
ArrayList<UInt>()

list.add(element)

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<K, MutableList<UInt>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(element)
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<K, MutableList<UInt>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(element)
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<K, MutableList<UByte>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(element)
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<K, MutableList<UInt>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(element)
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K, valueTransform: (UInt) -> V): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(valueTransform(element))
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K, valueTransform: (ULong) -> V): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(valueTransform(element))
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<K, MutableList<V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(valueTransform(element))
}

return destination
}

*/
@sinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K, valueTransform: (UShort) -> V): M {

for (element in this) {
    val key = keySelector(element)
    val list = destination.getOrPut(key)
    list.add(valueTransform(element))
}

return destination
}
```

Open Source Used In DNAC Platform Arica 12197
```kotlin
valueTransform: (UByte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array in * by the key returned by the given [keySelector] function applied to the element in *
 * and puts to the
 * [destination] map each group key associated with a list of corresponding values in *
 * @param The [destination] map.
 * @param * @sample samples.collections.Collections.Transformations.groupByKeysAndValues
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K, valueTransform: (UShort) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function in *
 * to each element in the original array.
 * @param * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.map(transform: (UInt) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function in *
 * to each element in the original array.
 * @param * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.map(transform: (ULong) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function in *
 * to each element in the original array.
 * @param * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function in *
 * to each element in the original array.
 * @param * @sample samples.collections.Collections.Transformations.map
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

---

*Open Source Used In DNAC Platform Arica 12198*
[transform] function that takes the index of an element and the element itself * and returns the result of the transform applied to the element.

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UIntArray.mapIndexedTo(destination: C, transform: (Int, UInt) -> R): C {
  var index = 0
  for (item in this)
    destination.add(transform(index++, item))
  return destination
}
```

Applies the given [transform] function to each element and its index in the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> ULongArray.mapIndexedTo(destination: C, transform: (Int, ULong) -> R): C {
  var index = 0
  for (item in this)
    destination.add(transform(index++, item))
  return destination
}
```

Applies the given [transform] function to each element and its index in the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UByteArray.mapIndexedTo(destination: C, transform: (Int, UByte) -> R): C {
  var index = 0
  for (item in this)
    destination.add(transform(index++, item))
  return destination
}
```

Applies the given [transform] function to each element and its index in the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C : MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (Int, UShort) -> R): C {
  var index = 0
  for (item in this)
    destination.add(transform(index++, item))
  return destination
}
```

Applies the given [transform] function to each element and its index in the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

Applies the given [transform] function to each element of the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

Applies the given [transform] function to each element of the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

Applies the given [transform] function to each element of the original array * and appends the results to the given [destination].

```kotlin
*\@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, C :MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}
```

Applies the given [transform] function to each element of the original array * and appends the results to the given [destination].
Iterable<IndexedValue<UByte>> { return IndexingIterable { iterator() } }\n
/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the\n * element itself.\n */
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.withIndex():\n    Iterable<IndexedValue<UByte>> {\n        return IndexingIterable { iterator() }\n    }\n
/**
 * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.all(predicate: (UInt) -> Boolean): Boolean \n    return storage.any() && predicate(first)\n
/**
 * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean \n    return storage.any() && predicate(first)\n
/**
 * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean \n    return storage.any() && predicate(first)\n
/**
 * Returns `true` if all elements match the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.all\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean \n    return storage.any() && predicate(first)\n
/**
 * Returns `true` if at least one element matches\n * the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.anyWithPredicate(predicate: (UInt) -> Boolean): Boolean \n    return storage.any() || predicate(first)\n
/**
 * Returns `true` if at least one element matches\n * the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.anyWithPredicate(predicate: (ULong) -> Boolean): Boolean \n    return storage.any() || predicate(first)\n
/**
 * Returns `true` if at least one element matches\n * the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.anyWithPredicate(predicate: (UByte) -> Boolean): Boolean \n    return storage.any() || predicate(first)\n
/**
 * Returns `true` if at least one element matches\n * the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.anyWithPredicate\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.anyWithPredicate(predicate: (UShort) -> Boolean): Boolean \n    return storage.any() || predicate(first)\n
/**
 * Returns the number of elements matching the given [predicate].\n * \n * @sample samples.collections.Collections.Aggregates.count\n */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic var UIntArray.count: Int get() = storage.count()
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun

UIntArray.count(predicate: (UInt) -> Boolean): Int {\n    var count = 0n
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun

ULongArray.count(predicate: (ULong) -> Boolean): Int {\n    var count = 0n
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun

UByteArray.count(predicate: (UByte) -> Boolean): Int {
    var count = 0n
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun

UShortArray.count(predicate: (UShort) -> Boolean): Int {
    var count = 0n
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>

UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\\npublic inline fun <R>
accumulator

**\* Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UShortArray.foldRightIndexed(initial: R, operation: (Int, acc: R, UShort) -> R): R {\n    var index = 0\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(index++, accumulator, element)\n    }\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(get(index--), accumulator)\n    }\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n    }\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UByteArray.foldRightIndexed(initial: R, operation: (Int, UByte, acc: R) -> R): R {\n    var index = lastIndex\n    var accumulator = initial\n    while (index >= 0) {\n        accumulator = operation(index, get(index), accumulator)\n        --index\n    }\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UShortArray.foldRightIndexed(initial: R, operation: (Int, acc: R, UShort) -> R): R {\n    var index = 0\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UShortArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UShort) -> R): R {\n    var index = 0\n    for (element in this) accumulator = initial\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UIntArray.foldIndexed(initial: R, operation: (Int, acc: R, UInt) -> R): R {\n    var index = 0\n    for (element in this) accumulator = operation(index++, acc, element)\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> ULongArray.foldIndexed(initial: R, operation: (Int, acc: R, ULong) -> R): R {\n    var index = 0\n    for (element in this) accumulator = operation(index++, acc, element)\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UByteArray.foldIndexed(initial: R, operation: (Int, acc: R, UByte) -> R): R {\n    var index = 0\n    for (element in this) accumulator = operation(index++, acc, element)\n    return accumulator\n}

**\* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element with its index in the original array.**

*\@SinceKotlin("1.3\")\*\@ExperimentalUnsignedTypes*\@kotlin.internal.InlineOnly*\public inline fun \<R> UShortArray.foldIndexed(initial: R, operation: (Int, acc: R, UShort) -> R): R {\n    var index = 0\n    for (element in this) accumulator = operation(index++, acc, element)\n    return accumulator\n}
accumulator
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself\n * and current accumulator value, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
    UShortArray.foldRightIndexed(initial: R, operation: (Int, UShort, acc: R) -> R): R
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.forEachIndexed(action: (Int, UInt) -> Unit): Unit
    var index = 0
    for (item in this) action(index++, item)

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.forEachIndexed(action: (Int, ULong) -> Unit): Unit
    var index = 0
    for (item in this) action(index++, item)

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.forEachIndexed(action: (Int, UByte) -> Unit): Unit
    var index = 0
    for (item in this) action(index++, item)

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * @param [action] function that takes the index of an element and the element itself\n * and performs the desired action on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.forEachIndexed(action: (Int, UShort) -> Unit): Unit
    var index = 0
    for (item in this) action(index++, item)

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.max(): UInt? {
    if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (max < e) max = e
    return max}
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.max(): ULong? {
    if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (max < e) max = e
    return max}
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.max(): UByte? {
    if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (max < e) max = e
    return max}
}

/**
 * Returns the largest element or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.max(): UShort? {
    if (isEmpty()) return null
    var max = this[0] for (i in 1..lastIndex) { val e = this[i] if (max < e) max = e
    return max}
}
fun UShortArray.max(): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

fun UIntArray.maxBy(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

fun ULongArray.maxBy(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UByteArray.maxBy(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UShortArray.maxBy(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UIntArray.min(): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

fun ULongArray.minBy(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

fun UByteArray.minBy(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

fun UShortArray.minBy(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

fun UIntArray.maxWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun ULongArray.maxWith(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.min(): ULong? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\nfunction UByteArray.min(): UByte? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.min(): UShort? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (min > e) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minBy(selector: (UInt) -> R): ULong? {\n    \n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minBy(selector: (ULong) -> R): ULong? {\n    \n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minBy(selector: (UByte) -> R): UByte? {\n    \n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minBy(selector: (UShort) -> R): UShort? {\n    \n    if (isEmpty()) return null\n    var minElem = this[0]\n    val lastIndex = this.lastIndex\n    if (lastIndex == 0) return minElem\n    var minValue = selector(minElem)\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        val v = selector(e)\n        if (minValue > v) {\n            minElem = e\n            minValue = v\n        }\n    }\n    return minElem\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.minWith(comparator: Comparator<in ULong>): ULong? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.minWith(comparator: Comparator<in UByte>): UByte? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}\n
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {\n    \n    if (isEmpty()) return null\n    var min = this[0]\n    for (i in 1..lastIndex) {\n        val e = this[i]\n        if (comparator.compare(min, e) > 0) min = e\n    }\n    return min\n}
Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minWith(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.minWith(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.minWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
```

```kotlin
UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRightIndexed(operation: (index: Int, UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexed(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRightIndexed(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
* to each element with its index in the original array and current accumulator value.
 * @param [operation]
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

Open Source Used In DNAC Platform Arica 12208
accumulator)
        --index
    }
    return accumulator
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.sumBy(selector: (ULong) -> ULong): ULong {
    var sum: ULong = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.sumBy(selector: (UByte) -> UByte): ULong {
    var sum: ULong = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.sumBy(selector: (UShort) -> UShort): ULong {
    var sum: ULong = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public infix fun <R> UByteArray.zip(other: Array<out R>): List<Pair<UByte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public infix fun <R> UShortArray.zip(other: Array<out R>): List<Pair<UShort, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.sumByDouble(selector: (UInt) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.sumByDouble(selector: (ULong) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.sumByDouble(selector: (UByte) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.sumByDouble(selector: (UShort) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
Array<out R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }\n} /* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index\n* using the provided [transform] function applied to each pair of elements. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list\n} /* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index\n* using the provided [transform] function applied to each pair of elements. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list\n} /* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index\n* using the provided [transform] function applied to each pair of elements. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UShortArray.zip(other: Array<out R>, transform: (a: UShort, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list\n} /* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index\n* using the provided [transform] function applied to each pair of elements. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list\n} /* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index\n* using the provided [transform] function applied to each pair of elements. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterableWithTransform

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun <R> UIntArray.zip(other: Iterable<R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }\n} /* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun <R> ULongArray.zip(other: Iterable<R>): List<Pair<ULong, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }\n} /* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun <R> UByteArray.zip(other: Iterable<R>): List<Pair<UByte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }\n} /* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterable

@SinceKotlin("1.3")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic infix fun <R> UShortArray.zip(other: Iterable<R>): List<Pair<UShort, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }\n} /* Returns a list of pairs built from the elements of ‘this’ collection and [other] array with the same index. \n* The returned list has length of the shortest collection. \n* \n* @sample samples.collections.Iterables.Operations.zipIterable
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@in\@kotlin.internal.InlineOnly\@public infix inline fun <R, V> UIntArray.zip\(other: Iterable\(R\)\)\(\to V\)\): List\(\langle V \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{val arraySize = size}\n    \text{val list = ArrayList\(V\)\(\langle\minOf(\text{other.collectionSizeOrDefault(10), arraySize})\rangle\)}
    \text{var i = 0 for (element in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n    }}
    \text{return list}\n}\right] *

Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterableWithTransform\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@in\@kotlin.internal.InlineOnly\@public inline fun <R, V> ULongArray.zip\(other: Iterable\(R\), transform:\(a: ULong, b: R\)\(\to V\)\): List\(\langle V \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{val arraySize = size}\n    \text{val list = ArrayList\(V\)\(\langle\minOf(\text{other.collectionSizeOrDefault(10), arraySize})\rangle\)}
    \text{var i = 0 for (element in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n    }}
    \text{return list}\n}\right] *

Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n * using the provided [transform] function applied to each pair of elements.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterableWithTransform\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@in\@kotlin.internal.InlineOnly\@public infix inline fun <R, V> UByteArray.zip\(other: Iterable\(R\), transform:\(a: UByte, b: R\)\(\to V\)\): List\(\langle V \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{val arraySize = size}\n    \text{val list = ArrayList\(V\)\(\langle\minOf(\text{other.collectionSizeOrDefault(10), arraySize})\rangle\)}
    \text{var i = 0 for (element in other) {\n    if (i >= arraySize) break\n    list.add(transform(this[i++], element))\n    }}
    \text{return list}\n}\right] *

Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterable\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@public infix fun UIntArray.zip\(other: UIntArray\): List\(\langle Pair<UInt, UInt> \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{return zip\(other\) \{ t1, t2 -> t1 to t2 \} }\right] *

Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterable\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@public infix fun ULongArray.zip\(other: ULongArray\): List\(\langle Pair<ULong, ULong> \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{return zip\(other\) \{ t1, t2 -> t1 to t2 \} }\right] *

Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterable\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@public infix fun UByteArray.zip\(other: UByteArray\): List\(\langle Pair<UByte, UByte> \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{return zip\(other\) \{ t1, t2 -> t1 to t2 \} }\right] *

Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.\n * * The returned list has length of the shortest collection.\n * * * @sample samples.collections.Operations.zipIterable\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@public infix fun UShortArray.zip\(other: UShortArray\): List\(\langle Pair<UShort, UShort> \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{return zip\(other\) \{ t1, t2 -> t1 to t2 \} }\right] *

Returns a list of values built from the elements of `this` array and the [other] array with the same index using the provided [transform] function applied to each pair of elements.\n * * The returned list has length of the shortest array.\n * * * @sample samples.collections.Operations.zipIterableWithTransform\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@in\@kotlin.internal.InlineOnly\@public inline fun <V> UIntArray.zip\(other: UIntArray, transform:\(a: UInt, b: UInt\)\(\to V\)\): List\(\langle V \rangle\) {
    \(\left[\begin{array}{l}
    \n    \text{val size = minOf(size, other.size)}\n    \text{val list = ArrayList\(V\)\(\langle size\)}
    \text{for (i in 0 until size) {\n    list.add(transform(this[i], other[i]))\n    }}
    \text{return list}\n}\right] *
/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @param other the other array
 * @param transform the function to apply to each pair of elements
 * @return a list containing the transformed values
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, W>

ULongArray.zip(other: ULongArray, transform: (a: ULong, b: ULong) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @param other the other array
 * @param transform the function to apply to each pair of elements
 * @return a list containing the transformed values
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, W>

UByteArray.zip(other: UByteArray, transform: (a: UByte, b: UByte) -> V): List<V> {
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @param other the other array
 * @param transform the function to apply to each pair of elements
 * @return a list containing the transformed values
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <V, W>

    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun 

Array<out UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun 

Array<out ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun 

Array<out UByte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun 

Array<out UShort>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

UIntArray.sum(): Int {
    return storage.sum().toUInt()
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

ULongArray.sum(): Int {
    return storage.sum().toULong()
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

UByteArray.sum(): Int {
    return sumBy { it.toUInt() }
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

UShortArray.sum(): Int {
    return sumBy { it.toUInt() }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @param other the other array
 * @param transform the function to apply to each pair of elements
 * @return a list containing the transformed values
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

samples.collections.Operations.zipIterableWithTransform

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest array.
 * @param other the other array
 * @param transform the function to apply to each pair of elements
 * @return a list containing the transformed values
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun 

samples.collections.Operations.zipIterableWithTransform
elements of this collection.

```kotlin
Collection<UByte>.toUByteArray(): UByteArray
```

```kotlin
Collection<UInt>.toUIntArray(): UIntArray
```

```kotlin
Collection<ULong>.toULongArray(): ULongArray
```

```kotlin
Collection<UShort>.toUShortArray(): UShortArray
```

```kotlin
Iterable<UInt>.sum(): UInt
```

```kotlin
Iterable<ULong>.sum(): ULong
```

```kotlin
Iterable<UByte>.sum(): UInt
```

```kotlin
Iterable<UShort>.sum(): UInt
```

```kotlin
maxOf(a: UInt, b: UInt): UInt
```

```kotlin
maxOf(a: ULong, b: ULong): ULong
```

```kotlin
maxOf(a: UByte, b: UByte): UByte
```

```kotlin
maxOf(a: UShort, b: UShort): UShort
```

```kotlin
minOf(a: UInt, b: UInt): UInt
```

```kotlin
minOf(a: ULong, b: ULong): ULong
```

```kotlin
minOf(a: UByte, b: UByte): UByte
```

```kotlin
minOf(a: UShort, b: UShort): UShort
```

```kotlin
@file:kotlin.jvm.JvmMultifileClass
```

```kotlin
@file:kotlin.jvm.JvmName("UComparisonsKt")
```

```kotlin
package kotlin.comparisons
```

```kotlin
//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
// https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt): UInt {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun maxOf(a: UByte, b: UByte, c: UByte): UByte {
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return maxOf(maxOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun minOf(a: UInt, b: UInt): UInt {
    return if (a <= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun minOf(a: UByte, b: UByte): UByte {
    return minOf(minOf(a, b), c)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun minOf(a: UShort, b: UShort): UShort {
    return minOf(minOf(a, b), c)
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 *
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UComparisonsKt")

package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
// https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("URangesKt")

package kotlin.ranges

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntRange.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongRange.random(): ULong {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntRange.random(random: Random): UInt {
    try {
        return random.nextUInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongRange.random(random: Random): ULong {
    try {
        return random.nextULong(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Checks if the specified value belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: UByte): Boolean {
    return contains(value.toUInt())
}

/**
 * Checks if the specified value belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: UByte): Boolean {
    return contains(value.toULong())
}

/*
 */

/*! @SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n\nfun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}\n
/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UByte, b: UByte): UByte {
    return if (a <= b) a else b
}\n
/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UShort, b: UShort): UShort {
    return if (a <= b) a else b
}\n
/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UInt, b: UInt, c: UInt): UInt {
    return minOf(a, minOf(b, c))
}\n
/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: ULong, b: ULong, c: ULong): ULong {
    return minOf(a, minOf(b, c))
}\n
/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(a, minOf(b, c))
}\n
/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return minOf(a, minOf(b, c))
}\n
/*! */
*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic operator fun UIntRange.contains(value: ULong): Boolean {\n    return (value shr Int.SIZE_BITS) == 0L && contains(value.toUInt())\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic operator fun UIntRange.contains(value: UShort): Boolean {
    return contains(value.toUInt())\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic operator fun ULongRange.contains(value: UByte): Boolean {
    return contains(value.toULong())\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULong.downTo(to: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(this, to, -1L)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UShort.downTo(to: UShort): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun UShort.downTo(to: UShort): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun ULong.downTo(to: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(this, to, -1L)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun UShort.downTo(to: UShort): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.reversed(): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.step(step: Int): UIntProgression {
    checkStepIsPositive(step > 0, step)\n    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.step(step: Long): ULongProgression {
    checkStepIsPositive(step > 0, step)\n    return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.reversed(): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.step(step: Int): UIntProgression {
    checkStepIsPositive(step > 0, step)\n    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.step(step: Long): ULongProgression {
    checkStepIsPositive(step > 0, step)\n    return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.reversed(): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun UIntProgression.step(step: Int): UIntProgression {
    checkStepIsPositive(step > 0, step)\n    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}

*\@SinceKotlin("1.3")*\n@ExperimentalUnsignedTypes*\npublic infix fun ULongProgression.step(step: Long): ULongProgression {
    checkStepIsPositive(step > 0, step)\n    return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n}
range is empty.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UShort.until(to: UShort): UIntRange {
    return UIntRange.EMPTY
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * 
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtLeast(minimumValue: UInt): UInt {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * 
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtLeast(minimumValue: ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * 
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtLeastUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtLeast(minimumValue: UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtMost(maximumValue: UInt): UInt {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtMost(maximumValue: ULong): ULong {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * 
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue] or the [maximumValue] otherwise.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    return if (this < minimumValue) minimumValue else if (this > maximumValue) maximumValue else this
}

/*
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {
    return if (minimumValue > maximumValue) maximumValue else this
}

/*
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {
    return if (minimumValue > maximumValue) maximumValue else this
}

/*
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {
    return if (minimumValue > maximumValue) maximumValue else this
}

/*
 * @SinceKotlin("1.3")
 * @ExperimentalUnsignedTypes
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {
    return if (minimumValue > maximumValue) maximumValue else this
}
minimumValue) return minimumValue
  if (this > maximumValue) return maximumValue
  return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue)
        return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue)
        return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue)
        return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(range: ClosedRange<UInt>): UInt {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<UInt>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * 
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(range: ClosedRange<ULong>): ULong {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<ULong>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}


/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("USequencesKt")

package kotlin.sequences

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
Sequence<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License,
 * Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS"
 * BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.collections

import kotlin.internal.PlatformDependent

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that can
 * be iterated over.
 * @param T the type of element being iterated over. The iterator is covariant on its element type.
 */
public interface Iterable<out T> {
    /**
     * Returns an iterator over the elements of this object.
     */
    public operator fun iterator(): Iterator<T>
}

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that can
 * be iterated over and that supports removing elements during iteration.
 * @param T the type of element being iterated over. The mutable iterator is invariant on its element type.
 */
public interface MutableIterable<out T> : Iterable<T> {
    /**
     * Returns an iterator over the elements of this sequence that supports removing elements during iteration.
     */
    override fun iterator(): MutableIterator<T>
}

/**
 * A generic collection of elements. Methods in this interface support only read-only access to the collection;
 * read/write access is supported through the
 * [MutableCollection] interface.
 * @param E the type of elements contained in the collection. The collection is covariant on its element type.
 */
public interface Collection<out E> : Iterable<E> {
    // Query Operations
    /**
     * Returns the size of the collection.
     */
    public val size: Int

    /**
     * Returns `true` if the collection is empty (contains no elements), `false` otherwise.
     */
    public fun isEmpty(): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Adds all of the elements of the specified collection to this collection.
     * @return `true` if any of the specified elements was added to the collection, `false` if the collection was not modified.
     */
    public fun addAll(elements: Collection<E>): Boolean
support duplicate elements, and supports `@param E the type of elements contained in the set. The mutable set is invariant on its element type.

```java
Set<E>, MutableCollection<E> { in // Query Operations
override fun iterator(): MutableIterator<E>
/**
 * Modifies the map by using the specified key/value pairs from another map.
 * `null` is returned as the replacement value for the `null` key.
 * @param from Another mutable map that provides key/value pairs
 * @return the number of key/value pairs added
 */
override fun putAll(from: Map<K, V>): Int
```

```java
/**
 * Adds the specified key to the set.
 * @param element Added element
 * @return `true` if the element has been added, `false` if the element is already contained in the set
 */
override fun add(element: E): Boolean
```

```java
/**
 * Removes the specified key from the set.
 * @param key Key to be removed
 * @return `true` if the key was present in the set, `false` if not
 */
override fun remove(key: K): Boolean
```

```java
/**
 * Removes all elements from this set.
 */
public fun clear(): Unit
```

```java
/**
 * Removes all elements specified by the `elements` collection.
 * @param elements Collection of elements to be removed
 * @return `true` if this set contained any of the specified elements
 */
override fun removeAll(elements: Collection<E>): Boolean
```

```java
/**
 * Removes the specified element from the set.
 * @param element Element to be removed
 * @return `true` if the element is contained in the set
 */
override fun remove(element: E): Boolean
```

```java
/**
 * Returns a read-only set of all keys in this map.
 */
public val keys: Set<K>
```

```java
/**
 * Returns a read-only collection of all values in this map.
 * @return a Read-only collection of all values in this map
 */
public val values: Collection<V>
```

```java
/**
 * Returns a read-only set of all key/value pairs in this map.
 */
public val entries: Set<Map.Entry<K, V>>
```

```java
/**
 * Returns a read-only collection of all key/value pairs in this map.
 * @return a read-only collection of all key/value pairs in this map
 */
public val keysAndValues: Collection<MutablePair<K, V>>
```

```java
/**
 * Returns a read-only collection of all key/value pairs in this map.
 * @return a read-only collection of all key/value pairs in this map
 */
public val keyValues: Collection<MutableMap.Entry<K, V>>
```

```java
/**
 * Returns a read-only collection of all key/value pairs in this map.
 * @return a read-only collection of all key/value pairs in this map
 */
public val entrySet: Collection<MutableMap.Entry<K, V>>
```

```java
/**
 * Returns `true` if the map contains a key that is mapped to the specified value.
 * @param value Value to check for presence
 * @return `true` if the map contains a key that is mapped to the specified value
 */
public fun containsValue(value: V): Boolean
```

```java
/**
 * Returns `true` if the map contains the specified key.
 * @param key Key to check for presence
 * @return `true` if the map contains the specified key
 */
public fun containsKey(key: K): Boolean
```

```java
/**
 * Removes all elements from this map.
 */
public fun clear(): Unit
```

```java
/**
 * Removes all elements from this map.
 */
public fun clear(): Unit
```

```java
/**
 * Associates the specified key with the specified value.
 * @param key Key to associate with the value
 * @param value Value to associate with the key
 */
public fun put(key: K, value: V): V
```

```java
/**
 * Associates the specified key/value pairs from another map.
 * @param from Another map
 */
public fun putAll(from: Map<K, V>): Int
```

```java
/**
 * Associates the specified key/value pairs from another map.
 * @param from Another map
 */
public fun putAll(from: Map<K, V>): Int
```

```java
/**
 * Associates the specified key with the specified value.
 * @param key Key to associate with the value
 * @param value Value to associate with the key
 */
public fun put(key: K, value: V): V
```

```java
/**
 * Associates the specified key/value pairs from another map.
 * @param from Another map
 */
public fun putAll(from: Map<K, V>): Int
```

```java
/**
 * Associates the specified key/value pairs from another map.
 * @param from Another map
 */
public fun putAll(from: Map<K, V>): Int
```
Open Source Used In DNAC Platform Arica 12221

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. 

DO NOT EDIT!\n
apackage kotlin.collections\n
/** An iterator over a sequence of values of type `Byte`. */
@public abstract class ByteIterator : Iterator<Byte> {\n    \n    override final fun next() = nextByte()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextByte(): Byte\n}

/** An iterator over a sequence of values of type `Int`. */
@public abstract class IntIterator : Iterator<Int> {\n    \n    override final fun next() = nextInt()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextInt(): Int\n}

/** An iterator over a sequence of values of type `Long`. */
@public abstract class LongIterator : Iterator<Long> {\n    \n    override final fun next() = nextLong()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextLong(): Long\n}

/** An iterator over a sequence of values of type `Boolean`. */
@public abstract class BooleanIterator : Iterator<Boolean> {\n    \n    override final fun next() = nextBoolean()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextBoolean(): Boolean\n}

/** An iterator over a sequence of values of type `Double`. */
@public abstract class DoubleIterator : Iterator<Double> {\n    \n    override final fun next() = nextDouble()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextDouble(): Double\n}

/** An iterator over a sequence of values of type `Float`. */
@public abstract class FloatIterator : Iterator<Float> {\n    \n    override final fun next() = nextFloat()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextFloat(): Float\n}

/** An iterator over a sequence of values of type `Char`. */
@public abstract class CharIterator : Iterator<Char> {\n    \n    override final fun next() = nextChar()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextChar(): Char\n}

/** An iterator over a sequence of values of type `Byte`. */
@public abstract class ByteIterator : Iterator<Byte> {\n    \n    override final fun next() = nextByte()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextByte(): Byte\n}

/** An iterator over a sequence of values of type `Char`. */
@public abstract class CharIterator : Iterator<Char> {\n    \n    override final fun next() = nextChar()\n/** Returns the next value in the sequence without boxing. */
\n    public abstract fun nextChar(): Char\n}
override fun hashCode(): Int =
    if (isEmpty()) -1 else (31 * (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))) + (step xor (step ushr 32))).toInt()

    override fun toString(): String = if (step > 0)
        "$first..$last step $step"
    else "$first downTo $last step ${-step}"

    companion object {
        /**
         * Creates LongProgression within the specified bounds of a closed range.
         * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].
         * In order to go backwards the [step] must be negative.
         * [step] must be greater than `Long.MIN_VALUE` and not equal to zero.
         */
        public fun fromClosedRange(rangeStart: Long, rangeEnd: Long, step: Long): LongProgression = LongProgression(rangeStart, rangeEnd, step)
    }
}

/*
 * Copyright 2010-2019
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.ranges

/**
 * Represents a range of values (for example, numbers or characters).
 * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/ranges.html) for more information.
 */
public interface ClosedRange<T: Comparable<T>> {
    /**
     * The minimum value in the range.
     */
    public val start: T

    /**
     * The maximum value in the range (inclusive).
     */
    public val endInclusive: T

    /**
     * Checks whether the specified [value] belongs to the range.
     */
    public operator fun contains(value: T):

    /**
     * Checks whether the range is empty.
     */
    public fun isEmpty(): Boolean = start > endInclusive
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin

/**
 * The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java.
 */
public object Unit {
    override fun toString() = "kotlin.Unit"
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.annotation

import kotlin.annotation.AnnotationTarget.*

/**
 * Contains the list of code elements which are the possible annotation targets.
 */
public enum class AnnotationTarget {
    /**
     * Class, interface or object, annotation class is also included
     */
    CLASS,
    /**
     * Annotation class only
     */
    ANNOTATION_CLASS,
    /**
     * Generic type parameter (unsupported yet)
     */
    TYPE_PARAMETER,
    /**
     * Property
     */
    PROPERTY,
    /**
     * Field, including property's backing field
     */
    FIELD,
    /**
     * Local variable
     */
    LOCAL_VARIABLE,
    /**
     * Value parameter of a function or a constructor
     */
    VALUE_PARAMETER,
    /**
     * Constructor only (primary or secondary)
     */
    CONSTRUCTOR,
    /**
     * Function (constructors are not included)
     */
    FUNCTION,
    /**
     * Property getter only
     */
    PROPERTY_GETTER,
    /**
     * Type property setter only
     */
    PROPERTY_SETTER,
    /**
     * Any expression
     */
    TYPE,
    /**
     * File alias
     */
    FILE,
    /**
     * Type alias
     */
    TYPEALIAS,
    /**
     *Determines how an annotation is stored in binary output.
     */
    TYPE_ALIAS
}

/**
 * Contains the list of possible annotation's retentions.
 */
public enum class AnnotationRetention {
    /**
     * Annotation isn't stored in binary output
     */
    SOURCE,
    /**
     * Annotation is stored in binary output, but visible for reflection
     */
    RUNTIME,
    /**
     * Annotation is stored in binary output and visible for reflection (default retention)
     */
    RETENTION
}

Open Source Used In DNAC Platform Arica 12223
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 * No validation on passed parameters is performed. The given parameters should satisfy the condition:
 * - either [step] > 0 and [start] <= [end],
 * - or [step] < 0 and [start] >= [end].
 * @param start first element of the progression
 * @param end ending bound for the progression
 * @param step increment, or difference of successive elements in the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end)
        end else end + differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.internal

private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 * No validation on passed parameters is performed. The given parameters should satisfy the condition:
 * - either [step] > 0 and [start] <= [end],
 * - or [step] < 0 and [start] >= [end].
 * @param start first element of the progression
 * @param end ending bound for the progression
 * @param step increment, or difference of successive elements in the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end)
        end else end + differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.internal

private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 * No validation on passed parameters is performed. The given parameters should satisfy the condition:
 * - either [step] > 0 and [start] <= [end],
 * - or [step] < 0 and [start] >= [end].
 * @param start first element of the progression
 * @param end ending bound for the progression
 * @param step increment, or difference of successive elements in the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end)
        end else end + differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}
```kotlin
* @suppress
*/
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long):
    Long = when {
        step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
        step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
        else -> throw kotlin.IllegalArgumentException("Step is zero.")
    }

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in
 * compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 * either express or implied. See the License for the specific language
 * governing permissions and limitations under the License.
 */

package kotlin.reflect

/**
 * Represents a parameter passed to a function or a property getter/setter,
 * including 'this' and extension receiver parameters.
 */
public interface KParameter : KAnnotatedElement {
    /**
     * 0-based index of this parameter in the
     * parameter list of its containing callable.
     */
    public val index: Int

    /**
     * Name of this parameter as it
     * was declared in the source code,
     * or 'null' if the parameter has no name or its name is not available at
     * runtime.
     * Examples of nameless parameters include 'this' instance for member functions,
     * extension receiver for extension functions or properties, parameters of Java methods,
     * compiled without the debug
     * information, and others.
     */
    public val name: String?

    /**
     * Type of this parameter. For a 'vararg'
     * parameter, this is the type of the corresponding array.
     */
    public val type: KType

    /**
     * Kind of this parameter.
     */
    public enum class Kind {
        /** Instance required to make a call to
         * the member, or an outer class instance for an inner class constructor.
         */
        INSTANCE,
        /** Extension
         * receiver of an extension function or property.
         */
        EXTENSION_RECEIVER,
        /** Ordinary named
         * parameter.
         */
        VALUE,
    }

    /**
     * `true` if this parameter is optional and can be omitted when
     * making a call via [KCallable.callBy], or `false` otherwise.
     * A parameter is optional in any of the two cases:
     * 1. The default value is provided at the declaration of this parameter.
     * 2. The parameter is declared in a member function and one of the corresponding parameters in the super functions is optional.
     * `true` if this parameter is a 'vararg' parameter.
     * * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/functions.html#variable-number-of-arguments-varargs)
     */
    public val isOptional: Boolean

    /**
     * `true` if this parameter is 'vararg'.
     * * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/functions.html#variable-number-of-arguments-varargs)
     */
    @SinceKotlin("1.1")
    public val isVararg: Boolean
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * Represents a type. Type is usually either a class with optional type arguments,
 * or a type parameter of some
 * declaration, plus nullability. Type arguments passed for the parameters of the classifier in this type.
 * For example, in the type `List<String>` the classifier would be the [KClass] instance for [List].
 */
public interface KType : KAnnotatedElement {
    /**
     * The declaration of
     * the classifier used in this type.
     * For example, in the type `Outer<out Number>` the only type argument is `out Number`.
     * * In case this type is based on an inner class, the
     *   returned list contains the type arguments provided for the innermost class first, then
     *   its outer class, and so on.
     * * For example, in the type `Array<out A, B>` the returned list is `A, B`.
     * For Kotlin types, it means that 'null' value is allowed to be
     * represented by this type.
     */
    public val classifier: KClassifier?

    /**
     * Type arguments passed for the parameters of the classifier in this type.
     * For example, in the type `Array<out Number>` the only type argument is `out Number`.
     * * In case this type is based on an inner class, the
     *   returned list contains the type arguments provided for the innermost class first, then its outer class, and so on.
     * For example, in the type `Array<out A, B>` the returned list is `A, B`.
     * For Kotlin types, it means that 'null' value is allowed to be
     * represented by this type.
     * In practice it means that the type was declared with a question mark at the end.
     * * For non-Kotlin types, it means the type or the symbol which was declared with this type
     *   is annotated with a runtime-retained nullability annotation such as [java.annotation.Nullable].
     *   Note that even if [isMarkedNullable] is false, values of the type can still be 'null'.
     * This may happen if it is a type of the type parameter with a nullable upper bound.
     */
    @SinceKotlin("1.1")
    public val arguments: List<KTypeProjection>

    /**
     * `true` if this type was marked nullable in the source code.
     */
    public val isMarkedNullable: Boolean

    /**
     * `true` if this type is a 'vararg'.
     */
    @SinceKotlin("1.1")
    public val isVararg: Boolean
}
```

Open Source Used In DNAC Platform Arica  12225
false for t's type, but t can be null here when T = `\Any\`.

public val isMarkedNullable: Boolean

/**
 * Represents a type projection. Type projection is usually the argument to another type in a
 * type usage. The use-site variance specified in the projection, or `null` if this is a star projection.
 * The type specified in the projection, or `null` if this is a star projection.
 * Star projection, denoted by the `*` character. For example, in the type `\Array\<\out Number\>`.
 * `\out Number` is the covariant projection of the type represented by the class `\Number\`.
 * Type projection is either the star projection, or an entity consisting of a specific type plus optional variance.
 * See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#type-projections) for more information.
 */

/**
 * Represents variance applied to a type parameter on the declaration site ("declaration-site variance"),
 * or to a type in a projection ("use-site variance"). See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/generics.html#variance) for more information.
 */

@SinceKotlin("1.1")
enum class KVariance {
    /**
     * The affected type parameter or type is invariant, which means it has no variance
     * applied to it.
     */
    INVARIANT,

    /**
     * The affected type parameter or type is contravariant. Denoted by the `in` modifier in the source code.
     */
    IN,

    /**
     * The affected type parameter or type is covariant. Denoted by the `out` modifier in the source code.
     */
    OUT,
}

/**
 * Copyright 2010-2016 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */

package kotlin.reflect

/**
 *Visibility is an aspect of a Kotlin declaration regulating where that declaration is accessible in the source code.
 *Visibility can be changed with one of the following modifiers: `public`, `protected`, `internal`, `private`.
 *Note that some Java visibilities such as package-private and protected (which also gives access to items from the same package)
 *cannot be represented in Kotlin, so there's no `KVisibility` value corresponding to them.
 */

@SinceKotlin("1.1")
enum class KVisibility {
    /**
     * Represents value applied to a type parameter on the declaration site (*declaration-site variance*), or to a type in a projection (*use-site variance*).
     */
    DECLARATION_SITE,

    /**
     * The affected type parameter or type is `invariant`, which means it has no variance
     * applied to it.
     */
    INVARIANT,

    /**
     * The affected type parameter or type is `contravariant`
     * Denoted by the `in` modifier in the source code.
     */
    CONTRAVARIANT,

    /**
     * The affected type parameter or type is `covariant`
     * Denoted by the `out` modifier in the source code.
     */
    COVARIANT,
}

/**
 * Copyright 2010-2016 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */

package kotlin.reflect

// ...
Since Kotlin("1.1")
enum class KVisibility {
    /**
     * Visibility of declarations marked with the `public` modifier, or with no modifier at all.
     */
    PUBLIC,

    /**
     * Visibility of declarations marked with the `protected` modifier.
     */
    PROTECTED,

    /**
     * Visibility of declarations marked with the `internal` modifier.
     */
    INTERNAL,

    /**
     * Visibility of declarations marked with the `private` modifier.
     */
    PRIVATE,
}

fun arrayIterator(array: dynamic, type: String?) = when (type) {
    null -> {
        val arr: Array<dynamic> = array
        object : Iterator<dynamic> {
            var index = 0
            override fun hasNext() = index < arr.size
            override fun next() = if (index < arr.size) arr[index++] else throw NoSuchElementException("$index")
        }
    }
    "BooleanArray" -> booleanArrayIterator(array)
    "ByteArray" -> byteArrayIterator(array)
    "ShortArray" -> shortArrayIterator(array)
    "CharArray" -> charArrayIterator(array)
    "IntArray" -> intArrayIterator(array)
    "LongArray" -> longArrayIterator(array)
    "FloatArray" -> floatArrayIterator(array)
    "DoubleArray" -> doubleArrayIterator(array)
    else -> throw IllegalStateException("Unsupported type argument for arrayIterator: $type")
}

fun booleanArrayIterator(array: BooleanArray) = object : BooleanIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextBoolean() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun byteArrayIterator(array: ByteArray) = object : ByteIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextByte() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun shortArrayIterator(array: ShortArray) = object : ShortIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextShort() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun charArrayIterator(array: CharArray) = object : CharIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextChar() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun intArrayIterator(array: IntArray) = object : IntIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextInt() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun floatArrayIterator(array: FloatArray) = object : FloatIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextFloat() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun doubleArrayIterator(array: DoubleArray) = object : DoubleIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextDouble() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun longArrayIterator(array: LongArray) = object : LongIterator {
    var index = 0
    override fun hasNext() = index < array.size
    override fun nextLong() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun PropertyMetadata(@JsName("callableName") val name: String) = object {
    override fun hasNext() = index < array.size
    override fun next() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")
}

fun propertyMetadata(@JsName("newThrowable") val message: String?, cause: Throwable?):
Throwable { val throwable = js("new Error()")
throwable.message = if (jsTypeOf(message) == "undefined") {
   if (cause != null) cause.toString() else null
} else {
   message
}
throwable.cause = cause
throwable.name = "Throwable"
return
}

internal class BoxedChar(val c: Int) : Comparable<Int> {
    override
    fun equals(other: Any?): Boolean {
        return other is BoxedChar && c == other.c
    }
    override
    fun hashCode(): Int {
        return c
    }
    override
    fun toString(): String {
        return js("this.c").unsafeCast<Char>().toString()
    }
    override
    fun compareTo(other: Int): Int {
        return this.c - other.c
    }
}
A discriminated union that encapsulates successful outcome with a value of type [T]\n or a failure with an arbitrary [Throwable] exception.\n
```kotlin
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")\npublic inline class Result<out T> @PublishedApi internal constructor(\n    @PublishedApi\n    internal val value: @JvmField\n    val exception: Throwable\n) : Serializable {
    // discovery

    /**
     * Returns an instance that encapsulates the given [value] as successful value.\n     * In this case [isSuccess] returns `true`.\n     * In this case [getValue] returns [value]\n     * when (value) {\n         if (value is Failure) throw value.exception
     }\n
    /**
     * Returns a string `Success(v)` if this instance represents successful outcome.\n     * In this case [isSuccess] returns `true`.\n     */
    public fun exceptionOrNull(): Throwable? =
        when (value) {
            is Failure -> throw value
            else -> null
        }

    /**
     * Returns an instance that encapsulates the given [value] as successful value.\n     * In this case [isSuccess] returns `true`.\n     */
    public val isSuccess: Boolean get() = value !is Failure

    // value & exception

    /**
     * Returns an instance that encapsulates the given [value] as successful value.\n     * In this case [isSuccess] returns `true`.\n     */
    public fun getValue(): T? =
        when {
            isFailure -> null
            else -> value as T
        }

    /**
     * Returns an instance that encapsulates the given [value] as successful value.\n     * In this case [isSuccess] returns `true`.\n     */
    override fun toString(): String = "Success($value)"
}
```

The above code snippet is an implementation of the `Result` class in Kotlin, which is used to represent either a successful outcome with a value of type `T` or a failure with an arbitrary `Exception`. It provides methods to check the success or failure of the result, get the value, and convert the result to a string. The `Result` class is a discriminated union that serves as a simple way to work with successful and failed outcomes in a functional programming style.
`getOrElse { throw it }` (see [getOrElse]).

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.getOrThrow(): T {
    throwOnFailure()
    return value as T
}
```

```kotlin
/**
 * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the
 * result of [onFailure] function for encapsulated exception if it is [failure][Result.isFailure].\n * Note, that an exception thrown by [onFailure] function is rethrown by this function.\n * This function is shorthand for `fold(onSuccess = { it }, onFailure = onFailure)` (see [fold]).\n */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.getOrElse(onFailure: (exception: Throwable) -> R): R {
    contract {
        callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null ->
        else -> onFailure(exception)
    }
}
```

```kotlin
/**
 * Returns the encapsulated value if this instance represents [success][Result.isSuccess] or the
 * [defaultValue] if it is [failure][Result.isFailure].\n * This function is shorthand for `getOrElse { defaultValue }` (see [getOrElse]).\n */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.getOrDefault(defaultValue: R): R {
    if (isFailure) return defaultValue
    return value as T
}
```

```kotlin
/**
 * Returns the result of [onSuccess] for encapsulated value if this instance represents [success][Result.isSuccess]
 * or the result of [onFailure] function for encapsulated exception if it is [failure][Result.isFailure].
 * Note, that an exception thrown by [onSuccess] or [onFailure] function is rethrown by this function.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.fold(
    onSuccess: (value: T) -> R,
    onFailure: (exception: Throwable) -> R
): R {
    contract {
        callsInPlace(onSuccess, InvocationKind.AT_MOST_ONCE)
        callsInPlace(onFailure,
            InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> onSuccess(value as T)
        else -> onFailure(exception)
    }
}
```

```kotlin
/**
 * Returns the encapsulated result of the given [transform] function applied to encapsulated value\n * if this instance represents [success][Result.isSuccess] or the\n * original encapsulated exception if it is [failure][Result.isFailure].\n * Note, that an exception thrown by [transform] function is rethrown by this function.\n * See [mapCatching] for an alternative that encapsulates exceptions.\n */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.map(transform: (value: T) -> R): Result<R> {
    contract {
        callsInPlace(transform, InvocationKind.AT_MOST_ONCE)
    }
    return when {
        isSuccess -> Result.success(transform(value as T))
        else -> Result(value)
    }
}
```

```kotlin
/**
 * Returns the encapsulated result of the given [transform] function applied to encapsulated exception\n * if this instance represents [success][Result.isSuccess] or the\n * original encapsulated exception if it is [failure][Result.isFailure].\n * Any exception thrown by [transform] function is caught, encapsulated as a failure and returned by this function.\n * See [map] for an alternative that rethrows exceptions.\n */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.mapCatching(transform: (value: T) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { transform(exception) }
    }
}
```

```kotlin
/**
 * Returns the encapsulated result of the given [transform] function applied to encapsulated exception\n * if this instance represents [success][Result.isSuccess] or the\n * original encapsulated exception if it is [failure][Result.isFailure].\n * Any exception thrown by [transform] function is rethrown by this function.\n */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.recover(transform: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { transform(exception) }
    }
}
```

```kotlin
/**
 * Performs the given [action] on encapsulated exception
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onFailure(action: Throwable.() -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { exception.() }
    }
}
```

```kotlin
/**
 * Performs the given [action] on encapsulated exception
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onSuccess(action: T.() -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { exception.() }
    }
}
```
if this instance represents [failure][Result.isFailure].

*\n@InlineOnly\n@SinceKotlin("1.3")\npublic inline fun \(<T>\ Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)\n    }\n    exceptionOrNull()?.let { action(it) }\n    return this
}\

*\n@InlineOnly\n@SinceKotlin("1.3")\npublic inline fun \(<T>\ Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    if (isSuccess)
        action(value as T)
    return this
}

+-------------------+
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be
 * found in the license/LICENSE.txt file.
 */

package kotlin.coroutines

import kotlin.coroutines.intrinsics.*
import kotlin.internal.InlineOnly

/**
 * Interface representing a continuation after
 * a suspension point that returns a value of type `T`.
 */
@SinceKotlin("1.3")
public interface Continuation<in T> {
    /**
     * The context of the coroutine that corresponds to this continuation.
     */
    public val context: CoroutineContext

    /**
     * Resumes the execution of the corresponding coroutine passing a successful or
     * failed [result] as the return value of the last suspension point.
     */
    public fun resumeWith(result: Result<T>)
}

/**
 * Resumes the execution of the corresponding coroutine so that the [exception] is re-thrown right after the
 * last suspension
 * point.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation<T>.resumeWithException(exception: Throwable): Unit =
    resumeWith(Result.failure(exception))

/**
 * Creates a [Continuation] instance with the given [context] and
 * implementation of [resumeWith] method.
 */
@SinceKotlin("1.3")
@InlineOnly\npublic inline fun <T> Continuation(
    context: CoroutineContext,
    crossinline resumeWith: (Result<T>) -> Unit
): Continuation<T> =
    object : Continuation<T> {
        override val context: CoroutineContext
            get() = context

        override fun resumeWith(result: Result<T>) =
            resumeWith(result)
    }

/**
 * Creates a coroutine with receiver type [R] and result type [T].
 * This function creates a new, fresh instance of suspendable
 * computation every time it is invoked.
 */
@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun \(<T>\ R.() -> T>.createCoroutine(
    receiver: R,
    completion: Continuation<T>\n): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(completion).intercepted(), COROUTINE_SUSPENDED)\n        \n    *\n   ******/

*\n@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun \(<T>\ R.() -> T>.createCoroutine(
    receiver: R,
    completion: Continuation<T>\n): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(), COROUTINE_SUSPENDED)\n        \n    *\n    */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <T> (suspend () ->
T).startCoroutine(
    completion: Continuation<T>
) {
    createCoroutineUnintercepted(completion).intercepted().resume(Unit)
}

/**
 * Starts a coroutine with receiver type [R] and result type [T].
 * This function creates and starts a new, fresh instance of suspendable computation every time it is invoked.
 * The [completion] continuation is invoked when the coroutine completes with a result or an exception.
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <R, T> (suspend
R.() -> T).startCoroutine(
    receiver: R,
    completion: Continuation<T>
) {
    createCoroutineUnintercepted(receiver, completion).intercepted().resume(Unit)
}

/**
 * Obtains the current continuation instance inside suspend functions and suspends
 * the currently running coroutine.
 * In this function both [Continuation.resume] and [Continuation.resumeWithException] can be used either synchronously
 * in the same stack-frame where the suspension function is run or asynchronously later in the same thread or
 * from a different thread of execution. Subsequent invocation of any resume function will produce an
 * [IllegalStateException].
 */
@SinceKotlin("1.3")
@InlineOnly
public suspend inline fun <T> suspendCoroutine(crossinline block: (Continuation<T>) -> Unit): T =
    suspendCoroutineUninterceptedOrReturn {
        c: Continuation<T> ->
        val safe = SafeContinuation(c.intercepted())
        block(safe)
        safe.getOrThrow()
    }

/**
 * Returns the context of the current coroutine.
 */
@SinceKotlin("1.3")
@Suppress("WRONG_MODIFIER_TARGET")
@InlineOnly
public suspend inline
val coroutineContext: CoroutineContext
    get() {
        throw NotImplementedError("Implemented as intrinsic")
    }

/**
 * Starts an unintercepted coroutine without a receiver and with result type [T] and executes it until its first
 * suspension.
 */
@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <T> (suspend () ->
T).startCoroutineUninterceptedOrReturn(
    completion: Continuation<T>
): Any? =
    this.asDynamic()(completion, false)

/**
 * Starts an unintercepted coroutine with receiver type [R] and result type [T] and executes it until its first
 * suspension.
 */
@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <R, T> (suspend R.() ->
T).startCoroutineUninterceptedOrReturn(
    receiver: R,
    completion: Continuation<T>
): Any? =
    this.asDynamic()(receiver, false)

/**
 * Creates unintercepted coroutine without receiver and with
 * result type [T].
 */
@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <T> (suspend () ->
T).startCoroutineUninterceptedOrReturn(
    completion: Continuation<T>
): Any? =
    this.asDynamic()(completion, false)

/**
 * Creates unintercepted coroutine without receiver and with
 * result type [T].
 */
@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <R, T> (suspend R.() ->
T).startCoroutineUninterceptedOrReturn(
    receiver: R,
    completion: Continuation<T>
): Any? =
    this.asDynamic()(receiver, completion, false)
established.

* Note that [completion] of this function may get invoked in an arbitrary context.
* Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of `resume(Unit)` on intercepted continuation happens in the invocation context established by `supply(Unit)`.
* Repeated invocation of any resume function on the resulting continuation corrupts the state machine of the coroutine and may result in arbitrary behaviour or exception.

* `createCoroutineUnintercepted(Continuation<Void>)` is public actual function `<T>` (suspend `()->T`).

```kotlin
private inline fun <T> createCoroutineFromSuspendFunction(completion: Continuation<T>, crossinline block: () -> Any?): Continuation<Unit> {
    @Suppress("UNCHECKED_CAST")
    return object: CoroutineImpl(completion as Continuation<Any?>) {
        override fun doResume(): Any? {
            exception?.let { throw it }
            return block()
        }
    }
}
```

```kotlin
package kotlin.js

// Mirrors signature from JS IR
@library
@JsName("imulEmulated")
internal fun imul(x: Int, y: Int): Int = definedExternally

internal inline fun isArrayish(o: dynamic) =
    js("Kotlin").isArrayish(o)
```

```kotlin
@file:Suppress("UNUSED_PARAMETER")

package kotlin.js

@kotlin.internal.InlineOnly
internal inline fun jsDeleteProperty(obj: Any, property: Any) {
    js("delete obj[property]")
}

@kotlin.internal.InlineOnly
internal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =
    js("lhs | rhs").unsafeCast<Int>()
```

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.
 */

@file:Suppress("UNUSED_PARAMETER")

package kotlin.js

@kotlin.internal.InlineOnly
internal inline fun jsDeleteProperty(obj: Any, property: Any) {
    js("delete obj[property]")
}

@kotlin.internal.InlineOnly
internal inline fun jsBitwiseOr(lhs: Any?, rhs: Any?): Int =
    js("lhs | rhs").unsafeCast<Int>()
```
Open Source Used In DNAC Platform Arica

*package kotlin.math

/**
 * Returns this value with the sign bit same as of the
 * [sign] value.
 * @param sign the sign bit
 * @returns Returns a bit representation of the specified floating-point value as 
 * [Long] value
 * @since kotlin-'1.2'
 */
public actual fun Double.withSign(sign: Double): Double {
    val thisSignBit = js("Kotlin").doubleSignBit(this).unsafeCast<Int>()
    val newSignBit = js("Kotlin").doubleSignBit(sign).unsafeCast<Int>()
    return if (thisSignBit == newSignBit) this else -this
}

*package kotlin

/**
 * Returns a bit representation of the specified floating-point value as 
 * [Long] value
 * @since kotlin-'1.2'
 */
public actual fun Double.toBits(): Long = definedExternally

/**
 * Returns a bit representation of the specified floating-point value as 
 * [Int] value
 * preserving `NaN` values exact layout.
 * @since kotlin-'1.2'
 */
public actual fun Double.toRawBits(): Int = definedExternally

/**
 * Returns the 
 * [Double] value corresponding to a given bit representation.
 * @since kotlin-'1.2'
 */

internal inline fun Long(low: Int, high: Int) = js("Kotlin").Long.fromBits(low, high).unsafeCast<Long>()
internal inline val Long.low: Int get() = this.asDynamic().getLowBits().unsafeCast<Int>()
internal inline val Long.high: Int get() = this.asDynamic().getHighBits().unsafeCast<Int>()

*package kotlin

/**
 * Returns a string representation of this 
 * [Long] value in the specified 
 * [radix] base.
 * @param radix the radix
 * @returns Returns a string representation of this 
 * @since kotlin-'1.2'
 */
public actual fun Long.toString(radix: Int): String = asDynamic().toString(checkRadix(radix))

*package kotlin

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Reverses elements in the list in-place.
 */
public actual fun <T> MutableList<T>.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

*package kotlin

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Reverses elements in the list in-place.
 * @param list the list
 * @returns Returns a string representation of this 
 * @since kotlin-'1.2'
 */
public actual fun Long.toString(radix: Int): String = asDynamic().toString(checkRadix(radix))
sequence.

* @sample samples.collections.Collections.Elements.elementAt

* @public actual fun CharSequence.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") } \n} \n/**
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun CharSequence.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }
}

* @package collection.kt

* @sample kotlin.collections Collections\n* /NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
*/
See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib/\n* @nimport kotlin.collections.
* @nimport kotlin.ranges.contains
* @nimport kotlin.ranges.reversed
* * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
* * @sample samples.collections.Collections.Elements.elementAt

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun UIntArray.elementAt(index: Int): UInt {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun ULongArray.elementAt(index: Int): ULong {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun UByteArray.elementAt(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun UShortArray.elementAt(index: Int): UShort {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun UIntArray.asList(): List<UInt> {
    return object : AbstractList<UInt>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UInt): Boolean = this@asList.contains(element)
        override fun get(index: Int): UInt {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun ULongArray.asList(): List<ULong> {
    return object : AbstractList<ULong>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: ULong): Boolean = this@asList.contains(element)
        override fun get(index: Int): ULong {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes
* @public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
as Any?) !is UByte) return -1
                return this@asList.lastIndexOf(element)
            }
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 *
 * This may be useful in the following cases:
 *
 *   - There are two functions for which the compiler gives same name in JavaScript, you can
 *     mark one with `@JsName(...)` to prevent the compiler from reporting error.
 *   - You are writing a JavaScript library in Kotlin. The compiler produces mangled names
 *     for functions with parameters, which is unnatural for usual JavaScript developer.
 *
 * You can put `@JsName(...)` on functions you want to be available from JavaScript.
 */

//jquery UI
import jquery.JQuery
import kotlin.js.Json

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.buttonset(): JQuery = asDynamic().buttonset()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(): JQuery = asDynamic().dialog()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(mode: String, param: String, value: Any?): JQuery = asDynamic().dialog(mode, param, value)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.button(): JQuery = asDynamic().button()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.accordion(): JQuery = asDynamic().accordion()
hello() {
    println("Hello $name!")
}

fun hello(greeting: String) {
    println("$greeting $name!")
}

@JsName("helloWithGreeting")
fun hello() {
    println("Hello $name!")
}

@JsName(actual val name: String) *

/**
 * Denotes an `external` declaration that must be imported from native JavaScript library.
 * The compiler produces the code relevant for the target module system, for example, in case of
 * CommonJS, it will import the declaration via the `require(...)` function.
 * The annotation can be used on top-level external declarations (classes, properties, functions) and files.
 * In case of file (which can't be `external`) the following rule applies: all the declarations in
 * the file must be `external`. By applying the `@JsModule(...)` on a file you tell the compiler to import
 * a JavaScript object that contain all the declarations from the file.
 * Example:
 * ```
 * Hello world!
 * ```
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)
public actual annotation class JsName(actual val name: String)

/**
 * Denotes an `external` declaration that must be imported from native JavaScript library.
 * The compiler produces the code relevant for the target module system, for example, in case of
 * CommonJS, it will import the declaration via the `require(...)` function.
 * The annotation can be used on top-level external declarations (classes, properties, functions) and files.
 * In case of file (which can't be `external`) the following rule applies: all the declarations in
 * the file must be `external`. By applying the `@JsModule(...)` on a file you tell the compiler to import
 * a JavaScript object that contain all the declarations from the file.
 * Example:
 * ```
 * Hello world!
 * ```
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsModule(val import: String)

/**
 * Denotes an `external` declaration that can be used without module system.
 * By default, an `external` declaration is available regardless your target module system.
 * However, by applying the `@JsModule` annotation you can make a declaration unavailable to *plain* module system.
 * Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a
 * module available in the generated code. To tell the Kotlin compiler to accept both cases, you can augment
 * `@JsModule` with the `@JsNonModule` annotation.
 * ```
 * @file:JsQualifier("my.jsPackageName")
 * package some.kotlinPackage
 *
 * external fun foo(x: Int)
 *
 * external fun bar(): String
 * ```
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsNonModule

/**
 * Adds prefix to `external` declarations in a source file.
 * JavaScript does not have concept of packages (namespaces). They are usually emulated by nested objects.
 * The compiler turns references to `external` declarations either to plain unprefixed names (in case of *plain* modules) or to
 * plain imports. However, if a JavaScript library provides its declarations in packages, you won't be satisfied with
 * this. You can tell the compiler to generate additional prefix before references to `external` declarations using the
 * `@JsQualifier(...)` annotation. Note that a file marked with the `@JsQualifier(...)` annotation can't
 * contain non-`external` declarations. An example:
 * ```
 * package some.kotlinPackage
 *
 * external fun foo(x: Int): Int
 *
 * external fun bar(): String
 * ```
 */
@Retention(AnnotationRetention.BINARY)
@Target(FILE)
public annotation class JsQualifier(val value: String)

/**
 * Marks experimental JS export annotations.
 * Note that behaviour of these annotations will likely be changed in the future. Usages of such annotations will be reported as
 * warnings unless an explicit opt-in with the `@UseExperimental` annotation, e.g.
 * ```
 * @UseExperimental(ExperimentalJsExport::class)
 * ```
 * or with the `-XuseExperimental=kotlin.js.ExperimentalJsExport` compiler option is given.
 * Experimental.Level.WARNING
 */
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.FILE)
public annotation class ExperimentalJsExport

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0
license that can be found in the license/LICENSE.txt file.
 */

package kotlin.jvm

// these are used in common
generated code in stdlib

// TODO: find how to deprecate these

@Target(AnnotationTarget.FIELD)
@Retention(AnnotationRetention.SOURCE)
public actual
annotation class Volatile

@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER,
AnnotationTarget.PROPERTY_SETTER)
@Retention(AnnotationRetention.SOURCE)
public actual annotation
class Synchronized

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */

package kotlin.collections

/**
 * Provides a skeletal implementation of the
 * [MutableCollection] interface.
 *
 * @param E the type of elements contained in the collection. The collection is
 * invariant on its element type.
 */
public actual abstract class AbstractMutableCollection<E> protected actual
constructor() : AbstractCollection<E>(), MutableCollection<E> {

    actual abstract override fun add(element: E): Boolean

    actual override fun remove(element: E): Boolean {
        val iterator = iterator()
        while (iterator.hasNext()) {
            if (iterator.next() == element) {
                iterator.remove()
                return true
            }
        }
        return false
    }

    actual override fun addAll(elements: Collection<E>): Boolean {
        var modified = false
        for (element in elements) {
            if (add(element)) modified = true
        }
        return modified
    }

    actual override fun removeAll(elements: Collection<E>): Boolean = (this as
MutableIterable<E>).removeAll { it in elements }
    actual override fun retainAll(elements: Collection<E>): Boolean = (this as
MutableIterable<E>).removeAll { it !in elements }

    actual override fun clear(): Unit {
        val iterator = this.iterator()
        while (iterator.hasNext()) {
            iterator.next()
            iterator.remove()
        }
    }

    @JsName("toJSON")
    open fun toJSON(): Any = this.toArray()
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */
/*
 * Based on GWT AbstractList
 * Copyright 2007 Google Inc.
*/

package kotlin.collections

/**
 * Provides a skeletal implementation of
 * the [MutableList] interface.
 *
 * @param E the type of elements contained in the list. The list is invariant on its
 * element type.
 */
public actual abstract class AbstractMutableList<E> protected actual constructor() :
AbstractMutableCollection<E>(), MutableList<E> {
    protected var modCount: Int = 0

    abstract override fun
    add(index: Int, element: E): Unit
    abstract override fun
    removeAt(index: Int): E
    abstract override fun
    set(index: Int, element: E): E

    /**
     * Adds the specified element to the end of this list.
     *
     * @return `true` because the list is always modified as the result of this operation.
     */
    actual override fun add(element:
E): Boolean {
        add(size, element)
        return true
    }

    actual override fun addAll(index: Int, elements:
Collection<E>): Boolean {
        var _index = index
        var changed = false
        for (e in elements) {
            add(_index++, e)
            changed = true
        }
        return changed
    }

    actual override fun clear() {
        removeRange(0, size)
    }

    actual override fun removeAll(elements: Collection<E>): Boolean = removeAll {
        it in elements }
    actual override fun retainAll(elements: Collection<E>): Boolean = removeAll {
        it !in elements }

    actual override fun iterator(): MutableListIterator<E> = listIterator(0)
    actual override fun
    listIterator(index: Int): MutableListIterator<E> = ListIteratorImpl(index)

    actual override fun
    subList(fromIndex: Int, toIndex: Int): MutableList<E> = SubList(this, fromIndex, toIndex)

    /**
     * Removes the range of elements from this list
     * starting from [fromIndex] and ending with but not including [toIndex].
     */
    protected open fun
    removeRange(fromIndex: Int, toIndex: Int) {
        val iterator = listIterator(fromIndex)
        repeat(toIndex - fromIndex) {
            iterator.next()
            iterator.remove()
        }
    }

    /**
     * Compares this list with
     * another list instance with the ordered structural equality.
     */
    override fun equals(other: Any?): Boolean {
        if (other === this) return true
        if (other !is List<*>) return false

        return (other as List<*>).indexOf(this) >= 0
    }

    override fun hashCode(): Int = requireHashCode()}

Open Source Used In DNAC Platform Arica 12239
AbstractList.orderedEquals(this, other) \n} \n/** \n * Returns the hash code value for this list.\n */\noverride fun hashCode(): Int = AbstractList.orderedHashCode(this)\n
/**\n * Returns the hash code value for this list.\n */

class IteratorImpl :\n    MutableIterator<E> {
    /** the index of the item that will be returned on the next call to [next]() */\n    protected var index = 0\n    /** the index of the item that was returned on the previous call to [next]() or\n    [ListIterator.previous]() (for 'ListIterator'), -1 if no such item exists. */\n    protected var last = -1\n
    override fun hasNext(): Boolean = index < size\n
    override fun next(): E {
        if (!hasNext()) throw NoSuchElementException()\n        last = index++\n        return get(last)\n    }\n
    override fun removeAt(last: Int)\n    }\n
    override fun removeAt(last: Int): E {
        removeAt(last)\n        index = last\n        last = -1\n        }\n
    private open inner class IteratorImpl :\n        MutableListIterator<E> {

        init {
            AbstractList.checkPositionIndex(index,\n                  this@AbstractMutableList.size)\n            this.index = index\n        }\n
        override fun hasPrevious(): Boolean = index > 0\n
        override fun previous(): E {
            if (!hasPrevious()) throw NoSuchElementException()\n            last = --index\n            return get(last)\n        }\n
        override fun previousIndex(): Int = index - 1

        override fun add(index: Int, element: E) {
            add(index, element)\n            index++\n            last = -1\n        } \n
        override fun set(index: Int, element: E): E {
            return list.set(index, element)\n        }\n
        override val size: Int get() = _size
    }
}

package kotlin.collections

/**\n * Provides a skeletal implementation of the [MutableMap] interface.\n *\n * The implementor is required to implement [entries] property, which should\n * return mutable set of map entries, and\n * [put] function.\n *\n * @param K the type of map keys. The map is invariant on its key type.\n * @param V the type\n * of map values. The map is invariant on its value type.\n */\npublic actual abstract class AbstractMutableMap<K, V> {

        constructor(entry: Map.Entry<K, V>) :\n            this(entry.key, entry.value)\n
        private var _value = value\n
        override val value: V get() = _value

        override fun setValue(newValue: V): V {
            val oldValue = this._value\n            this._value = newValue\n            return oldValue
        }

        override fun hashCode(): Int = entryHashCode(this)\n        override fun toString(): String = entryToString(this)\n        override fun equals(other: Any?): Boolean = entryEquals(this, other)
    }

    actual override fun clear() {
        entries.clear()
    }

    private var _keys: MutableSet<K>? = null\n    actual override val keys: MutableSet<K> get() {
        if (_keys == null) \n            _keys = toIndex -\n
        override fun removeAt(index: Int): E {
            AbstractList.checkElementIndex(index,\n                  this@AbstractMutableMap.size)\n            list.removeAt(index)\n            _size--
            return result
        }

        override fun set(index: Int, element: E): E {
            return list.set(index, element)\n        }\n
        override val size: Int get() = _size
    }


Open Source Used In DNAC Platform Arica 12240
override fun hasNext(): Boolean = entryIterator.hasNext()
override fun remove(element: K): Boolean {
return true
}
...

override fun put(key: K, value: V): V? {
    return null
}

override fun putAll(from: Map<out K, V>) {
    for ((key, value) in from) {
        put(key, value)
    }
}

private var _values: MutableCollection<V>? = null
override val values: MutableCollection<V> get() {
    if (_values == null) {
        _values = object : AbstractMutableCollection<V>() {
            override fun add(element: V): Boolean = throw

            override fun clear() = this@AbstractMutableMap.clear()

            override operator fun contains(element: V): Boolean = containsValue(element)

            override operator fun iterator(): MutableIterator<V> {
                val

                override fun remove() = entryIterator.remove()
            }

            override val size: Int get() = this@AbstractMutableMap.size

            // TODO: should we implement them
            override fun equals(other: Any?): Boolean {
                if (this === other) return true
                if (other !is Collection<*>) return false
                return
            }

            override fun hashCode(): Int = AbstractList.orderedHashCode(this)
        }
        return _values!!
    }

    actual override fun remove(key: K): V? {
        val iter = entries.iterator()
        while (iter.hasNext()) {
            val entry =
            iter.next()
            val k = entry.key
            if (key == k) {
                val value = entry.value
                iter.remove()
                return value
            }
        }
        return null
    }

}

/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.collections

/**
 * Provides a skeletal implementation of the [MutableSet] interface.
 * @param E the type of elements contained
 * in the set. The set is invariant on its element type.
 */
public actual abstract class AbstractMutableSet<E> protected actual constructor() : AbstractMutableCollection<E>(), MutableSet<E> {

    /**
     * Compares this set with another set instance with the unordered structural equality.
     * @return `true`, if [other] instance is a [Set] of the same size, all elements of which are contained in this set.
     */
    override fun equals(other: Any?): Boolean {
        if (other === this) return true
        if (other !is Set<*>) return false
        return
    }

    /**
     * Returns the hash code value for this set.
     */
    override fun hashCode(): Int = AbstractSet.unorderedHashCode(this)

}
ensureCapacity(minCapacity: Int) { }

actual override val size: Int get() = array.size

@Suppress("UNCHECKED_CAST")
actual override fun get(index: Int): E = array[rangeCheck(index)] as E

actual override fun set(index: Int, element: E): E {
        rangeCheck(index)
        @Suppress("UNCHECKED_CAST")
        return array[index].apply { array[index] = element } as E
    }

actual override fun add(element: E): Boolean {
    if (elements.isEmpty()) return false
    array += elements.toTypedArray<Any?>()
    modCount++
    return true
}

actual override fun add(index: Int, element: E): Unit {
    array.asDynamic().splice(insertionRangeCheck(index), 0, element)
    modCount++
}

actual override fun addAll(elements: Collection<E>): Boolean {
    if (elements.isEmpty()) return false
    array += elements.toTypedArray<Any?>()
    modCount++
    return true
}

actual override fun addAll(index: Int, elements: Collection<E>): Boolean {
    insertionRangeCheck(index)
    if (index == size) return addAll(elements)
    if (elements.isEmpty()) return false
    when (index) {
        size -> return addAll(elements)
        0 -> array = elements.toTypedArray<Any?>() + array
        else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
    }
    modCount++
    return true
}

actual override fun removeAt(index: Int): E {
        rangeCheck(index)
        modCount++
        return if (index == lastIndex)
            array.asDynamic().pop()
        else
            array.asDynamic().splice(index, 1)[0]
    }

actual override fun remove(element: E): Boolean {
        for (index in array.indices) {
            if (array[index] == element) {
                array.asDynamic().splice(index, 1)
                modCount++
                return true
            }
        }
        return false
    }

override fun removeRange(fromIndex: Int, toIndex: Int) {
    modCount++
    array.asDynamic().splice(fromIndex, toIndex - fromIndex)
}

actual override fun clear() {
        array = emptyArray()
        modCount++
}

actual override fun indexOf(element: E): Int = array.indexOf(element)

actual override fun lastIndexOf(element: E): Int = array.lastIndexOf(element)

override fun toString() = arrayToString(array)

override fun toArray(): Array<Any?> = js("["]").slice.call(array)

private fun rangeCheck(index: Int) =

private fun insertionRangeCheck(index: Int) =

private fun sortArrayWith(array: Array<out T>, comparison: (T, T) -> Int) {
    if (getStableSortingIsSupported()) {
        array.asDynamic().sort(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, Comparator(comparison))
    }
}

private fun sortArrayWith(array: Array<out T>, comparator: Comparator<in T>) {
    if (getStableSortingIsSupported()) {
        val comparison = { a: T, b: T -> comparator.compare(a, b) }
        array.asDynamic().sort(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>(), 0, array.lastIndex, comparator)
    }
}

private fun sortArray(array: Array<out T>) {
    if (getStableSortingIsSupported()) {
        val comparison = { a: T, b: T -> a.compareTo(b) }
        array.asDynamic().sort(naturalOrder())
        return
    }
    _stableSortingIsSupported = null
    if (naturalOrder() == stableOrder) {
        var _stableSortingIsSupported = stableOrder
        if (stableOrder == naturalOrder())
            return
    }
    private fun mergeSort(array: Array<out T>, buffer: Array<T>, start: Int, end: Int, comparator: Comparator<in T>): Array<T>
        return
    }

private fun mergeSort(array: Array<out T>, buffer: Array<T>, start: Int, end: Int, comparator: Comparator<in T>): Array<T>
    if (start == end)
    }
val median = (start + end) / 2
val left = mergeSort(array, buffer, start, median, comparator)
val right = mergeSort(array, buffer, median + 1, end, comparator)
val target = if (left === buffer) array else buffer

var leftIndex = start
var rightIndex = median + 1
for (i in start..end) {
    var leftValue = left[leftIndex]
    var rightValue = right[rightIndex]
    if (comparator(leftValue, rightValue) <= 0) {
        target[i] = leftValue
        leftIndex++
    } else {
        target[i] = rightValue
        rightIndex++
    }
}

// Merge.
var leftIndex = start
var rightIndex = median + 1
for (i in start..end) {
    when {
        leftIndex <= median && rightIndex <= end -> {
            val leftValue = left[leftIndex]
            val rightValue = right[rightIndex]
            if (comparator(leftValue, rightValue) <= 0) {
                target[i] = leftValue
                leftIndex++
            } else {
                target[i] = rightValue
                rightIndex++
            }
        }
        leftIndex <= median -> {
            target[i] = left[leftIndex]
            leftIndex++
        }
        else /* rightIndex <= end */ -> {
            target[i] = right[rightIndex]
            rightIndex++
            Unit // TODO: Fix KT-31506
        }
    }
}

return target
}
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : this()

// This implementation of HashMap has no need of load factors or capacities.
require(initialCapacity >= 0) { "Negative initial capacity: $initialCapacity" }
require(loadFactor >= 0) { "Non-positive load factor: $loadFactor" }

actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

/**
 * Constructs an instance of [HashMap] filled with the contents of the specified [original] map.
 */
actual constructor(original: Map<out K, V>) : this()

this.putAll(original)

/**
 * Constructs the specialized implementation of [HashMap] with [String] keys, which stores the keys as properties of
 * JS object without hashing them.
 */
public fun stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
    return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply { putAll(pairs) }
}

/**
 * Creates a new instance of the specialized implementation of [HashSet] with the specified [String] elements,
 * which elements the keys as properties of JS object without hashing them.
 */
public fun stringSetOf(vararg elements: String): HashSet<String> {
    return HashSet(stringMapOf<Any>()).apply { addAll(elements) }
}
The insertion order is preserved by maintaining a doubly-linked list of all of its entries.

public actual open class LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {

    private class ChainEntry<K, V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V> {
        internal var next: ChainEntry<K, V>? = null
        internal var prev: ChainEntry<K, V>? = null
    }

    private inner class EntrySet : AbstractMutableSet<MutableEntry<K, V>>() {

        private inner class EntryIterator :MutableIterator<MutableEntry<K, V>> {
            // The last entry that was returned from this iterator.
            private var last: ChainEntry<K, V>? = null

            // The next entry to return from this iterator.
            private var next: ChainEntry<K, V>? = null

            init {
                next = head
            }

            override fun hasNext(): Boolean {
                return next !== null
            }

            override fun next(): MutableEntry<K, V> {
                if (!hasNext()) throw NoSuchElementException()

                val current = next!!
                last = current
                next = current.next.takeIf { it !== head }
                return current
            }

            override fun remove() {
                check(last != null)
                last!!.remove()
                map.remove(last.key)
                last = null
            }
        }

        override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on entries")

        override fun clear() {
            this@LinkedHashMap.clear()
        }

        override operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)

        override operator fun iterator():MutableIterator<MutableEntry<K, V>> = EntryIterator()

        override fun remove(element: MutableEntry<K, V>): Boolean {
            if (contains(element)) {
                this@LinkedHashMap.remove(element.key)
                return true
            }
            return false
        }

        override val size: Int get() = this@LinkedHashMap.size
    }

    /*
   * The head of the insert order chain, which is a doubly-linked circular
   * list. * the most recently inserted node is at the end of the chain, ie.
   * * private var var head: ChainEntry<K, V>?
    */
    private var head: ChainEntry<K, V>?

    /**
     * Add this node to the end of the chain.
     */
    private fun ChainEntry<K, V>.addToEnd() {
        // This entry is not in the list.
        check(next == null && prev == null)

        val _head = head
        if (_head == null) {
            head = this
            next = this
            prev = this
        } else {
            // Chain is
            val _tail = checkNotNull(_head.prev)
            // Update me.
            prev = _tail
            next = _head
            // Update my new siblings: current head and old tail
            _head.prev = this
            _tail.next = this
        }
    }

    /**
     * Remove this node from the chain it is a part of.
     */
    private fun ChainEntry<K, V>.remove() {
        if (this.next == this) {
            // if this is single element, remove head
            head = null
            else {
                if (head == this) {
                    // if this is first element, move head to next
                    head = next
                    prev = prev!!
                }
                this.prev.next = this
                this.next.prev = this
                if (_head == null) {
                    _head = this
                } else {
                    _head = next
                    _head.prev = this
                    prev = this
                }
            }
        } else {
            // Update this node from the chain it is a part of.
            this.prev.next = this
            this.next.prev = this
            if (_head == null) {
                _head = this
            } else {
                _head = next
                _head.prev = this
                prev = this
            }
        }
    }

    /*
   * The hashmap that keeps track of our entries and the chain. Note that we
   * duplicate the key here to eliminate changes to HashMap and minimize the
   */
    private val map: HashMap<K, ChainEntry<K, V>>

    /**
     * Constructs an empty
     */
    actual constructor() : super() {
        map = HashMap<K, ChainEntry<K, V>>()
    }

    internal constructor(backingMap: HashMap<K, Any>) {
        map = backingMap
    }

    override fun put(key: K, value: V): V {
        return putEntry(key, value)
    }

    override fun remove(key: K): V {
        return removeEntry(key).value
    }

    override fun containsKey(key: K): Boolean {
        return containsEntry(Entry(1, key))
    }

    override fun containsValue(value: V): Boolean {
        return containsEntry(Entry(2, value))
    }

    override fun size(): Int {
        return map.size
    }

    override fun isEmpty(): Boolean {
        return map.isEmpty()
    }

    override fun getOrDefault(key: K, defaultValue: V): V {
        return getEntry(key, defaultValue).value
    }

    override fun clear() {
        map.clear()
    }

    override fun equals(other: Any?): Boolean {
        return equals(other, map)
    }

    override fun hashCode(): Int {
        return hashCode(map)
    }

    override fun iterator(): Iterator<MutableEntry<K, V>> = EntrySet()
@Suppress("UNCHECKED_CAST") // expected to work due to erasure
        map = backingMap as
        HashMap<K, ChainEntry<K, V>>
    }

    /**
     * Constructs an empty 
     * [LinkedHashMap] instance.
     * @param initialCapacity the initial capacity (ignored)
     * @param loadFactor the load factor (ignored)
     * @throws IllegalArgumentException if the initial capacity or load factor are negative
     */
    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(initialCapacity, loadFactor) {
        map = HashMapK, ChainEntryK, V>(n) {
            map = HashMapK, ChainEntryK, V>(n) {
                head = null
            }
        }
        actual override fun clear() {
            map.clear()
            head = null
        }
        override fun clone(): Any
        actual override fun get(key: K): V? = map.get(key)
        actual override fun put(key: K, value: V): V? {
            old = map.get(key)
            if (old == null) {
                newEntry = ChainEntry(key, value)
                map.put(key, newEntry)
                newEntry.addToEnd()
                return null
            } else {
                old.setValue(value)
                return old
            }
        }
        actual override fun remove(key: K): V? {
            entry = map.remove(key)
            if (entry != null) {
                entry.remove(entry)
                return entry.value
            }
            return null
        }
        actual override val size: Int get() = map.size
    }

    /**
     * Constructs the specialized implementation of 
     * [LinkedHashMap] with [String] keys, which stores the keys as properties
     * of JS object without hashing them.
     */
    public fun linkedStringMapOf(vararg pairs: Pair<String, V>): LinkedHashMap<String, V> {
        return LinkedHashMap<Any>()).apply { putAll(pairs) }
    }

    /**
     * Creates a new instance of the specialized
     * implementation of [LinkedHashSet] with the specified [String] elements,
     * which elements the keys as properties
     * of JS object without hashing them.
     */
    public fun linkedStringSetOf(vararg elements: String): LinkedHashSet<String> {
        return LinkedHashSet(linkedStringMapOf<Any>()).apply { addAll(elements) }
    }

    @kotlin.internal.InlineOnly
    public fun
actual inline fun <R> synchronized(lock: Any, block: () -> R): R { // contract { \n    callsInPlace(block, InvocationKind.EXACTLY_ONCE) // \n    return block() // null \n    } /* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. \n    Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return block()
}

"/*
   * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
   * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
   */

package kotlin.io

internal abstract class BaseOutput {

    open fun println() {
        print("\n")
    }

    open fun println(message: Any?) {
        print(message)
        println()
    }

    abstract fun print(message: Any?)

    open fun flush() {}
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("NodeJsOutput")
internal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {
    override fun print(message: Any?) {
        val messageString = String(message)
        outputStream.write(messageString)
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("BufferedOutput")
internal open class BufferedOutput : BaseOutput() {
    var buffer = ""

    override fun print(message: Any?) {
        buffer += String(message)
    }

    override fun flush() {
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {

    override fun print(message: Any?) {
        var s = String(message)
        val i = s.lastIndexOf('\n')
        if (i >= 0) {
            buffer += s.substring(0, i)
            flush()
            s = s.substring(i + 1)
        }
        buffer += s
    }

    override fun flush() {
        console.log(buffer)
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("output")
internal var output = run {
    val isNode: Boolean = js("typeof process !== 'undefined' && process.versions && !process.versions.node")
    if (isNode) NodeJsOutput(js("process.stdout")) else BufferedOutputToConsoleLog()
}

@PublishedApi
@SinceKotlin("1.3")
internal actual class SafeContinuation<in T>
internal actual constructor(
    private val delegate: Continuation<T>,
    initialResult: Any?) : Continuation<T> {

    internal actual fun getOrThrow(): Any? {
        if (result === UNDECIDED) {
            result = COROUTINE_SUSPENDED // already called continuation, indicate COROUTINE_SUSPENDED upstream
        }
        val result = this.result
        return when {
            result === RESUMED ->
            result.value
            result is Result.Failure ->
            throw result
            else ->
            buffer += result
        }
    }

    override fun resumeWith(result: Result<T>) {
        this.result = result
    }

    override fun resumeWith(result: Result<T>, context: CoroutineContext) {
        this.result = result
    }
}

public actual fun println() {
    output.println()
}

public actual fun println(message: Any?) {
    output.println(message)
}

public actual fun print(message: Any?) {
    output.print(message)
}

license that can be found in the license/LICENSE.txt file.

```kotlin
package kotlin.dom

import org.w3c.dom.*

/**
 * Returns true if the element has the given CSS class style in its 'class' attribute
 * @fun Element.hasClass(cssClass: String): Boolean =
 * className.matches("(^|.*\s+)$cssClass($|\s+.*)".toRegex())
 *
 * @fun Element.addClass(vararg cssClasses: String): Boolean {
 *     val missingClasses = cssClasses.filterNot { hasClass(it) }
 *     if (missingClasses.isNotEmpty()) {
 *         val presentClasses = className.trim()
 *         className = buildString {
 *             append(presentClasses)
 *             if (!presentClasses.isEmpty()) {
 *                 append(" ")
 *             }
 *             missingClasses.joinTo(this, " ")
 *         }
 *         return true
 *     }
 *     return false
 * } * @fun Element.removeClass(vararg cssClasses: String): Boolean {
 *     if (cssClasses.any { hasClass(it) }) {
 *         val toBeRemoved = cssClasses.toSet()
 *         className = className.trim().split("\\s+".toRegex()).filter { it !in toBeRemoved }.joinTo(this, ",")
 *         return true
 *     }
 *     return false
 * }

/**
 * Converts the string into a regular expression [Regex] with the default options.
 * @fun String.toRegex(): Regex = Regex(this)
 * @fun String.toRegex(option: RegexOption): Regex = Regex(this, option)
 * @fun String.toRegex(options: Set<RegexOption>): Regex = Regex(this, options)

/**
 * Builds new string by populating newly created [StringBuilder] using provided
 * [builderAction] and then converting it to [String].
 * @fun buildString(builderAction: StringBuilder.() -> Unit): String =
 * StringBuilder().apply(builderAction).toString()
 * @fun buildString(capacity: Int, builderAction: StringBuilder.() -> Unit): String =
 * StringBuilder(capacity).apply(builderAction).toString()
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     for (item in value)
 *         append(item)
 *     return this
 * }

/**
 * Appends all arguments to the given [Appendable].
 * @fun Appendable.append<T: Appendable> T.append(vararg value: CharSequence?): T {
 *     for (item in value)
 *         append(item)
 *     return this
 * }

/**
 * Appends all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: String?): StringBuilder {
 *     for (item in value)
 *         append(item)
 *     return this
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(item: Any?): StringBuilder {
 *     append(item.toString())
 *     return this
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append<T: Appendable>(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: String): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: Any?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: CharSequence?): StringBuilder {
 *     return value.accumulate()
 * }

/**
 * Append all arguments to the given [StringBuilder].
 * @fun StringBuilder.append(vararg value: T): StringBuilder {
 *     return value.accumulate()
 * }*/
```

Open Source Used In DNAC Platform Arica 12251
this node is an [Element].

```kotlin
private class EventListenerHandler(private val handler: (Event) -> Unit) : EventListener {
    public override fun handleEvent(event: Event) {
        handler(event)
    }

    public override fun toString(): String = "EventListenerHandler($handler)"
}
```

```kotlin
public fun ItemArrayLike<T>.asList(): List<T> = object : AbstractList<T>() {
    override val size: Int get() = this@asList.length

    override fun get(index: Int): T = when (index) {
        in 0..lastIndex -> this@asList.item(index).unsafeCast<T>()
        else -> throw IndexOutOfBoundsException("index $index is not in range [0..$lastIndex]")
    }
}
```

```kotlin
fun Node.appendText(text: String): Element {
    appendChild(ownerDocument!!.createTextNode(text))
    return this
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun Any?.asDynamic(): dynamic = this

@kotlin.internal.InlineOnly
public inline fun <T> Any?.unsafeCast(): @kotlin.internal.NoInfer T = this.asDynamic()

@kotlin.internal.DynamicExtension
@JsName("unsafeCastDynamic")
@kotlin.internal.InlineOnly
public inline fun dynamic.unsafeCast(): @kotlin.internal.NoInfer T = this

@kotlin.internal.DynamicExtension
public operator fun dynamic.iterator(): Iterator<dynamic> {
    val r: Any? = this

    return when {
        this["iterator"] != null -> this["iterator"]() as T
        isArrayish(r) -> r.unsafeCast<Array<*>>().iterator()

        else -> (r as Iterable<*>).iterator()
    }
}
```

```kotlin
@kotlin.internal.NoInfer T = this.asDynamic()/null/** * Reinterprets this `dynamic` value as a value of the specified type [T] without any actual type checking.
*/

// TODO: remove workarounds for KT-22053 from
// TODO: remove primary constructors, make all secondary KT-
```
2055\n\n@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")\npublic actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor(): this(null, null) {
        Error::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        Error::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?) : this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor(): this(null, null) {
        Exception::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        Exception::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class RuntimeException actual constructor(message: String?, cause: Throwable?) : Exception(message, cause ?: null) {
    actual constructor(): this(null, null) {
        RuntimeException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        RuntimeException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        IllegalArgumentException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        IllegalArgumentException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        IllegalStateException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        IllegalStateException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        IndexOutOfBoundsException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        IndexOutOfBoundsException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class ConcurrentModificationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        ConcurrentModificationException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        ConcurrentModificationException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        UnsupportedOperationException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        UnsupportedOperationException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        NumberFormatException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        NumberFormatException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class NullPointerException actual constructor(message: String?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        NullPointerException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        NullPointerException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

public actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message, cause ?: null) {
    actual constructor(): this(null, null) {
        ClassCastException::class.js.asDynamic().call(this, null, null)
    }
    actual constructor(message: String?) : this(message, null) {
        ClassCastException::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?): this(cause)
    
    actual constructor(cause:Throwable?): this(cause)
    
    actual constructor() : this(null, null)
}

private actual constructor(message: String?, cause: Throwable?) : Error(message, cause) {
    actual constructor(): this(null)
    actual constructor(message: String?) : this(message)
    actual constructor(cause: Throwable?): this(cause)
}

actual constructor(message: String?) : this(message)

actual constructor() : this(message)

private constructor(message: Any?) :

// Since Kotlin("1.1")
public actual fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) { acc, _ -> acc + 1 }

/*
** Groups elements from the [Grouping] source by key and sums values provided by
** the [valueSelector] function for elements in each group.
** @return a [Map] associating the key of each group with the count of element in the group.
** @sample SinceKotlin("1.1")
*/
@SinceKotlin("1.1")
public inline fun <T, K> Grouping<T, K>.eachSumOf(valueSelector: (T) -> Int): Map<K, Int> =
        fold(0) { acc, e -> acc + valueSelector(e) }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.collections

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each
element to get its key.
 * A [Grouping] structure serves as an intermediate step in group-and-fold operations:
 * they group elements by their keys and then fold each group with some aggregating operation.
 * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
 * To get an instance of [Grouping] use one of `groupingBy` extension functions.
**
* Array
* CharSequence
* Collection
* Function
*/
@SinceKotlin("1.1")
public interface Grouping<T, out K> {
    /** Returns an [Iterator] over the elements of the source of this grouping. */
    fun sourceIterator(): Iterator<T>
    /** Extracts the key of an [element]. */
    fun keyOf(element: T): K
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group
 * sequentially, passing the previously accumulated value and the current element as arguments, and stores the
 * results in a new map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first element encountered in the group;
 *  - `element`: the element from the source being aggregated;
 *  - `first`: indicates whether it's the first element encountered in the group.
 * @return the [Map] associating the key of each group with the result of aggregation of the group elements.
 **
* @sample samples.collections.Grouping.aggregateByRadix
* @sample samples.collections.Grouping.aggregateByRadixTo
*/
@SinceKotlin("1.1")
public inline fun <T, K, R> Grouping<T, K>.aggregateTo(
    destination: MutableMap<K, R>,
    operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R
): MutableMap<K, R> {
    for (e in this.sourceIterator()) {
        val key = keyOf(e)
        val accumulator = destination[key]
        destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))
    }
    return destination
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation a function that is invoked on each element with the following parameters:
 *  - `key`: the key of the group this element belongs to;
 *  - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first element encountered in the group;
 *  - `element`: the element from the source being aggregated;
 *  - `first`: indicates whether it's the first element encountered in the group.
 * @return the [Map] associating the key of each group with the result of aggregation of the group elements.
 **
* @sample samples.collections.Grouping.aggregateByRadix
* @sample samples.collections.Grouping.aggregateByRadixTo
*/
@SinceKotlin("1.1")
public fun <T, K, R : Collection<T>, M : Map<K, R>> Grouping<T, K>.aggregateTo(
    destination: M,
    operation: (key: K, accumulator: M, element: T, first: Boolean) -> M
) { for (e in this.sourceIterator()) { val key = keyOf(e) val accumulator = destination[key] destination[key] = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key)) } return destination }
with the following parameters:

- `key`: the key of the group this element belongs to;
- `accumulator`: the current value of the accumulator of the group;
- `element`: the element from the source being accumulated.

* @return a [Map] associating the key of each group with the result of accumulating the group elements.

* @sample samples.collections.Grouping.foldByEvenLengthWithComputedInitialValue

* @SinceKotlin("1.1") * `public inline fun <T, K, R> Grouping<T, K>.fold(
initialValueSelector: (key: K, element: T) -> R,

@Suppress("UNCHECKED_CAST")
aggregate { key, acc, e, first -> operation(key, if (first)
initialValueSelector(key, e) else acc as R, e) } \n\n\n*Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.*

An initial value of accumulator is provided by [initialValueSelector] function.

* @param initialValueSelector a function that provides an initial value of accumulator for each group.

* @param operation a function that is invoked on each element with the following parameters:

  - `key`: the key of the group this element belongs to;
  - `accumulator`: the current value of the accumulator of the group;
  - `element`: the element from the source being accumulated.

* @return a [Map] associating the key of each group with the result of accumulating the group elements.

* @sample samples.collections.Grouping.foldByEvenLengthWithComputedInitialValueTo

* @SinceKotlin("1.1") * `public inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(

initialValueSelector: (key: K, element: T) -> R,

@Suppress("UNCHECKED_CAST")
aggregateTo(destination) { key, acc, e, first -> operation(if (first)
initialValueSelector(key, e) else acc as R, e) } \n\n\n*Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.*

* An initial value of accumulator is the same [initialValue] for each group.

* @param operation a function that is invoked on each element with the following parameters:

  - `accumulator`: the current value of the accumulator of the group;
  - `element`: the element from the source being accumulated.

* @return a [Map] associating the key of each group with the result of accumulating the group elements.

* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue

* @SinceKotlin("1.1") * `public inline fun <T, K, R> Grouping<T, K>.fold(

initialValue: R,

@Suppress("UNCHECKED_CAST")
aggregate { _, acc, e, first -> operation(if (first)
initialValue else acc as R, e) } \n\n\n*Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given [destination] map.*

An initial value of accumulator is the same [initialValue] for each group.

* @param operation a function that is invoked on each element with the following parameters:

  - `accumulator`: the current value of the accumulator of the group;
  - `element`: the element from the source being accumulated.

* @return a [Map] associating the key of each group with the result of accumulating the group elements.

* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValueTo

* @SinceKotlin("1.1") * `public inline fun <T, K, R, M : MutableMap<in K, R>> Grouping<T, K>.foldTo(

initialValue: R,
operation: (accumulator: R, element: T) -> R): M =

@Suppress("UNCHECKED_CAST")
aggregateTo(destination) { _, acc, e, first -> operation(if (first)
initialValue else acc as R, e) } \n\n\n*Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in a new map.*
An initial value of accumulator is the first element of the group.

@param operation a function that is invoked on each subsequent element of the group with the following parameters:

- `key`: the key of the group this element belongs to;
- `accumulator`: the current value of the accumulator of the group;
- `element`: the element from the source being accumulated.

@return a [Map] associating the key of each group with the result of accumulating the group elements.

@sample samples.collections.Grouping.reduceByMaxVowels

@SinceKotlin("1.1")
public inline fun <S, T : S, K> Grouping<T, K>.reduce(  
    operation: (key: K, accumulator: S, element: T) -> S
): Map<K, S> =
    aggregate { key, acc, e, first ->
        if (first) e else operation(key, acc as S, e)
    }

/**
 * Groups elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, passing the previously accumulated value and the
 * current element as arguments, and stores the results in the given [destination] map.
 * An initial value of accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group that value is used as an initial
 * value of the accumulator for that group and the first element of that group is also subjected to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * - `accumulator`: the current value of the accumulator of the group;
 * - `element`: the element from the source being folded;
 * @return a [Map] associating the key of each group with the result of accumulating the group elements.
 * @sample samples.collections.Grouping.reduceByMaxVowelsTo
 */
@SinceKotlin("1.1")
public inline fun <S, T : S, K, M : MutableMap<in K, S>> Grouping<T, K>.reduceTo(  
    destination: M,
    operation: (key: K, accumulator: S, element: T) -> S
): M =
    aggregateTo(destination) { key, acc, e, first ->
        if (first) e else operation(key, acc as S, e)
    }

/**
 * Groups elements from the [Grouping] source by key and counts elements in each group to the given [destination] map.
 * An initial value of counter for that group is the count of elements.
 * @return the [destination] map associating the key of each group with the count of elements in the group.
 * @sample samples.collections.Grouping.groupingByEachCount
 */
@SinceKotlin("1.1")
public fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachCountTo(destination: M): M =
    foldTo(destination, 0) { acc, _ -> acc + 1 }

// TODO: sum by long and by double overloads

        foldTo(destination, 0L) { acc, e -> acc + valueSelector(e)}

public inline fun <T, K> Grouping<T, K>.sumEachByLong(valueSelector: (T) -> Long): Map<K, Long> =
        fold(0L) { acc, e -> acc + valueSelector(e)}

public inline fun <T, K, M : MutableMap<in K, Double>> Grouping<T, K>.sumEachByDoubleTo(destination: M, valueSelector: (T) -> Double): M =
        foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e)}

public inline fun <T, K> Grouping<T, K>.sumEachByDouble(valueSelector: (T) -> Double): Map<K, Double> =
        fold(0.0) { acc, e -> acc + valueSelector(e)}
"kotlin.random.Random\"), level = DeprecationLevel.WARNING)\n  public fun random(): Double\n  @Deprecated("Use kotlin.math.abs instead.", ReplaceWith("abs(value)\", ", \"kotlin.math.abs\")\n  , level = DeprecationLevel.ERROR)\n  public fun abs(value: Double): Double\n  @Deprecated("Use kotlin.math.acos instead.", ReplaceWith("acos(value)\", ", \"kotlin.math.acos\")\n  , level = DeprecationLevel.ERROR)\n  public fun acos(value: Double): Double\n  @Deprecated("Use kotlin.math.atan instead.", ReplaceWith("atan(value)\", ", \"kotlin.math.atan\")\n  , level = DeprecationLevel.ERROR)\n  public fun atan(value: Double): Double\n  @Deprecated("Use kotlin.math.atan2 instead.", ReplaceWith("atan2(y, x)\", ", \"kotlin.math.atan2\")\n  , level = DeprecationLevel.ERROR)\n  public fun atan2(y: Double, x: Double): Double\n  @Deprecated("Use kotlin.math.cos instead.", ReplaceWith("cos(value)\", ", \"kotlin.math.cos\")\n  , level = DeprecationLevel.ERROR)\n  public fun cos(value: Double): Double\n  @Deprecated("Use kotlin.math.sin instead.", ReplaceWith("sin(value)\", ", \"kotlin.math.sin\")\n  , level = DeprecationLevel.ERROR)\n  public fun sin(value: Double): Double\n  @Deprecated("Use kotlin.math.exp instead.", ReplaceWith("exp(value)\", ", \"kotlin.math.exp\")\n  , level = DeprecationLevel.ERROR)\n  public fun exp(value: Double): Double\n  @PublishedApi\n  internal fun trunc(value: Number): Double\n  @PublishedApi\n  internal fun sign(value: Number): Double\n
  @PublishedApi\n  internal fun sinh(value: Double): Double\n  @PublishedApi\n  internal fun cosh(value: Double): Double\n  @PublishedApi\n  internal fun tanh(value: Double): Double\n  @PublishedApi\n  internal fun asinh(value: Double): Double\n  @PublishedApi\n  internal fun acosh(value: Double): Double\n  @PublishedApi\n  internal fun atanh(value: Double): Double\n
  @PublishedApi\n  internal fun hypot(x: Double, y: Double): Double\n  @PublishedApi\n  internal fun expm1(value: Double): Double\n  @PublishedApi\n  internal fun log10(value: Double): Double\n  @PublishedApi\n  internal fun log2(value: Double): Double\n  @PublishedApi\n  internal fun log1p(value: Double): Double\n
  @PublishedApi\n  internal fun clz32(value: Int): Int
}

/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION")
@Deprecated("Use minOf or kotlin.math.min instead", ReplaceWith("minOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long =
  if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION")
@Deprecated("Use maxOf or kotlin.math.max instead", ReplaceWith("maxOf(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long =
  if (a >= b) a else b

// Open Source Used In DNAC Platform Arica 12257
ReplaceWith("\"maxOf(a, b)\"", level = DeprecationLevel.ERROR)\npublic fun Math.max(a: Long, b: Long): Long = \nif (a >= b) a else b
*/

package kotlin.js

/**
 * An interface for indexing access to a collection of key-value pairs, where type of key is [String] and type of value is [Any?][Any].\n * Calls of the function will be translated to indexing operation (square brackets) on the receiver with [propertyName] as the argument.\n * E.g. for next code:\n * ```js\n * var a = \'abc\' ; \n * var b = \'def\' ; \n * var c = \'ghi\' ; \n * var map = {\n * a : 1, \n * b : 2, \n * c : 3\n * } ; \n * ```
 * will be generated:\n * ```js\n * function test(j, p) \n * \n * return j[p] ; \n * }\n * ```
 * Calls of the function will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].\n * ```js\n * function test(j, p, new_value) \n * \n * return j[p] = new_value ; \n * }\n * ```
 * Returns a simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.\n * */

@file:Suppress("NOT_DOCUMENTED")\npackage kotlin.math\n
import kotlin.internal.InlineOnly\nimport kotlin.js.Math as nativeMath\n
// region
================ Double Math ========================================

/** Computes the sine of the angle [x] given in radians.\n * Special cases:\n * - `sin(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun sin(x: Double): Double = nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians.\n * Special cases:\n * - `cos(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun cos(x: Double): Double = nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians.\n * Special cases:\n * - `tan(NaN|+Inf|-Inf)` is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun tan(x: Double): Double = nativeMath.tan(x)

/** Computes the arc sine of the value [x]; the returned value is an\n * angle in the range from `-PI/2` to `PI/2` radians.\n * Special cases:\n * - `asin(x)` is `NaN`, when `abs(x)` > 1 or x is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun asin(x: Double): Double = nativeMath.asin(x)

/** Computes the arc cosine of the value [x]; the returned value is an\n * angle in the range from `0` to `PI` radians.\n * Special cases:\n * - `acos(x)` is `NaN`, when `abs(x)` > 1 or x is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun acos(x: Double): Double = nativeMath.acos(x)

/** Computes the arc tangent of the value [x]; the returned value is an\n * angle in the range from `-PI/2` to `PI/2` radians.\n * Special cases:\n * - `atan(x)` is `NaN`\n */\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun atan(x: Double): Double =
nativeMath.atan(x)\(\text{atan}\) - Returns the angle \(\theta\) of the polar coordinates \((r, \theta)\) that corresponds to the rectangular coordinates \((x, y)\) by computing the arc tangent of the value \(y / x\). The returned value is an angle in the range from \(-\pi\) to \(\pi\) radians.\n
\[
\text{atan}(2(0.0, 0.0)) = 0.0
\]

\[
\text{atan}(1.0) = \pi / 4
\]

**nativeMath.atan2(y, x)** - Computes the hyperbolic sine of the value \(|y| / |x|\). The returned value is positive \(y\) such that \(\sinh(y) = y\).\n
\[
\text{atan2}(0.0, 0.0) = 0.0
\]

\[
\text{atan2}(1.0) = \pi / 2
\]

**nativeMath.atan2(y, x)** - Computes the hyperbolic cosine of the value \(|y| / |x|\). The returned value is positive \(y\) such that \(\cosh(y) = y^2 - 1\).\n
\[
\text{atan2}(0.0, 0.0) = 0.0
\]

\[
\text{atan2}(1.0) = \pi / 2
\]

**nativeMath.atan2(y, x)** - Computes the hyperbolic tangent of the value \(|y| / |x|\). The returned value is the reciprocal of the hyperbolic tangent of the value \(|y| / |x|\).\n
\[
\text{atan2}(0.0, 0.0) = 0.0
\]

\[
\text{atan2}(1.0) = 1.0
\]

**nativeMath.asinh(x)** - Computes the inverse hyperbolic sine of the value \(|x|\). The returned value is the positive \(y\) such that \(\sinh(y) = x\).\n
\[
\text{asinh}(0.0) = 0.0
\]

\[
\text{asinh}(+\infty) = +\infty
\]

**nativeMath.acosh(x)** - Computes the inverse hyperbolic cosine of the value \(|x|\). The returned value is the positive \(y\) such that \(\cosh(y) = x\).\n
\[
\text{acosh}(0.0) = \text{NaN}
\]

\[
\text{acosh}(+\infty) = +\infty
\]

**nativeMath.tanh(x)** - Computes the hyperbolic tangent of the value \(|x|\). The returned value is the value \(|x| / (\cosh(2x))\).\n
\[
\text{tanh}(0.0) = 0.0
\]

\[
\text{tanh}(+\infty) = 1.0
\]

**nativeMath.log(x, base)** - Computes the logarithm of the value \(|x|\) to the given \(\text{base}\).\n
\[
\text{log}(0.0, \text{NaN}) = \text{NaN}
\]

\[
\text{log}(+\infty, \text{NaN}) = +\infty
\]

**nativeMath.sqrt(x)** - Computes the square root of the value \(|x|\).\n
\[
\text{sqrt}(0.0) = 0.0
\]

\[
\text{sqrt}(+\infty) = +\infty
\]

**nativeMath.exp(x)** - Computes \(e^x\), where \(e\) is the base of the natural logarithm (2.718281828459045).\n
\[
\text{exp}(0.0) = 1.0
\]

\[
\text{exp}(+\infty) = +\infty
\]

**nativeMath.log1p(x)** - Computes \(\log(1 + x)\), where \(x\) is a small number.\n
\[
\text{log1p}(0.0) = 0.0
\]

**nativeMath.pow(x, y)** - Computes \(x^y\).\n
\[
\text{pow}(0.0, +\infty) = 0.0
\]

\[
\text{pow}(+\infty, 0.0) = +\infty
\]
for special cases.\n*\n@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun log10(x: Double): Double = nativeMath.log10(x).unsafeCast<Double>() // TODO: Remove unsafe cast after removing public js.math\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun log2(x: Double): Double = nativeMath.log2(x).unsafeCast<Double>()

/**
 * Computes `ln(x + 1)`.
 *
 * This function can be implemented to produce more precise result for \(x\) near zero.\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun ceil(x: Double): Double = nativeMath.ceil(x).unsafeCast<Double>()

/**
 * Rounds the given value \([x]\) to an integer towards positive infinity.\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun floor(x: Double): Double = nativeMath.floor(x).unsafeCast<Double>()

/**
 * Rounds the given value \([x]\) to an integer towards zero.\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun truncate(x: Double): Double = nativeMath.trunc(x)

/**
 * Rounds the given value \([x]\) towards the closest integer with ties rounded towards even integer.
 *
 * Special cases:
 *   - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun sign(x: Double): Double = nativeMath.sign(x)


/**
 * Returns the absolute value of the given value \([x]\).\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun abs(x: Double): Double = nativeMath.abs(x)

/**
 * Returns the sign of the given value \([x]\):\n *   - `-1.0` if the value is negative,
 *   - zero if the value is zero,
 *   - `1.0` if the value is positive\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline val sign: Double get() = nativeMath.sign(this)

/**
 * Returns this value with the sign bit same as of the sign value.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun Double.withSign(sign: Int): Double = this.withSign(sign.toDouble())

/**
 * Returns the ulp (unit in the last place) of this value.
 *
 * An ulp is a positive distance between this value and the next nearest \([Double]\) value larger in magnitude.\n*\n*\n
@SinceKotlin("1.2")\n@InlineOnly\npublic actual inline fun Double.withSign(sign: Int): Double = this.withSign(sign.toDouble())
Cases:
- `NaN.ulp` is `NaN`
- `x.ulp` is `+Inf` when `x` is `-Inf` or `-Inf`
- `0.0.ulp` is Double.MIN_VALUE
- `Double.POSITIVE_INFINITY` - `this.isNaN()`
- `Double.NEGATIVE_INFINITY` - `this == 0.0`
- `Double.MAX_VALUE` - `Double.fromBits(this.toRawBits() + 1)`

Returns the [Double] value nearest to this value in direction of positive infinity.
- * Special cases: `x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`
- `x.nextTowards(0)` == `x`
- `x.nextTowards(Double.MAX_VALUE)` == `Double.MAX_VALUE`
- `x.nextTowards(-Inf)` == `-Inf`

`x.nextUp()` == `x + Double.MIN_VALUE`

`x.nextDown()` == `x - Double.MIN_VALUE`

router.nextTowards(to: Double): Double = Double when `{ in when { this.isNaN() } | this == 0.0 | this == Double.MAX_VALUE | this == Double.MIN_VALUE | this == Double.NEGATIVE_INFINITY | this == Double.POSITIVE_INFINITY | this == 0.0 | this == Double.MIN_VALUE | this == Double.NEGATIVE_INFINITY | this == Double.POSITIVE_INFINITY } else { to }`

nativeMath.round(this).unsafeCast<Double>()

Float Math

```
Float Math

```

```
The returned value is an angle in the range from `-\pi` to `\pi` radians.

Special cases:
- `atan2(0.0, 0.0)` is `0.0`
- `-atan2(0.0, x)` is `0.0` for `x > 0`
- `atan2(-0.0, x)` is `-0.0` for `x > 0`
- `-atan2(y, +Inf)` is `0.0` for `0 < y < +Inf`
- `-atan2(-y, -Inf)` is `0.0` for `-Inf < y < 0`
- `atan2(y, -Inf)` is `\pi` for `0 < y < +Inf`
- `-atan2(y, +Inf)` is `-\pi` for `-Inf < y < 0`
- `-atan2(y, 0.0)` is `\pi/2` for `y > 0`
- `-atan2(y, 0.0)` is `-\pi/2` for `y < 0`
- `-atan2(+Inf, x)` is `\pi/2` for finite `x`
- `-atan2(-Inf, x)` is `-\pi/2` for finite `x`
- `atan2(NaN, x)` and `atan2(y, NaN)` is `NaN`

@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan2(y: Float, x: Float): Float = nativeMath.atan2(y.toDouble(), x.toDouble()).toFloat()

/**
 * Computes the hyperbolic cosine of the value [x].
 * Special cases:
 *   - `cosh(NaN)` is `NaN`
 *   - `cosh(+Inf)` is `+Inf`
 *   - `cosh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cosh(x: Float): Float = nativeMath.cosh(x.toDouble()).toFloat()

/**
 * Computes the hyperbolic tangent of the value [x].
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(+Inf)` is `1.0`
 *   - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tanh(x: Float): Float = nativeMath.tanh(x.toDouble()).toFloat()

/**
 * Computes the inverse hyperbolic sine of the value [x].
 * The returned value is positive `y` such that `sinh(y) == x`.
 * Special cases:
 *   - `asinh(NaN)` is `NaN`
 *   - `asinh(+Inf)` is `+Inf`
 *   - `asinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asinh(x: Float): Float = nativeMath.asinh(x.toDouble()).toFloat()
of the value \(x\).\n
```kotlin
@SinceKotlin("1.2."
@InlineOnly
public
```
this.toDouble().withSign(sign.toDouble()).toFloat()\n\n// endregion

// region ================ Integer Math

/**
 * Returns the absolute value of the given value \[n\].
 *
 * Special cases:
 *   - \(\text{abs}(\text{Int.MIN_VALUE})\) is \(\text{Int.MIN_VALUE}\) due to an overflow\n * @see absoluteValue extension property for \[\text{Int}\]
 */
@SinceKotlin("1.2")
public actual fun abs(n: Int): Int = if (n < 0) (-n or 0) else n

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun min(a: Int, b: Int): Int = nativeMath.min(a, b)

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun max(a: Int, b: Int): Int = nativeMath.max(a, b)

/**
 * Returns the absolute value of this value.
 *
 * Special cases:\n * - \(\text{abs}(\text{Int.MIN_VALUE})\) is \(\text{Int.MIN_VALUE}\) due to an overflow\n * @see abs function\n */
@SinceKotlin("1.2")
public actual inline val Int.absoluteValue: Int get() = abs(this)

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,\n *   - `0` if the value is zero,\n *   - `1` if the value is positive\n * @see sign function\n */
@SinceKotlin("1.2")
public actual val Int.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}

/**
 * Returns the absolute value of the given value \[n\].
 * @see abs function\n */
@SinceKotlin("1.2")
public actual fun abs(n: Long): Long = if (n < 0) -n else n

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
@Suppress("NOTHING_TO_INLINE")
public actual inline fun min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
@Suppress("NOTHING_TO_INLINE")
public actual inline fun max(a: Long, b: Long): Long = if (a >= b) a else b

/**
 * Returns the absolute value of this value.
 *
 * Special cases:\n * - \(\text{abs}(\text{Long.MIN_VALUE})\) is \(\text{Long.MIN_VALUE}\) due to an overflow\n * @see absoluteValue extension property for \[\text{Long}\]
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline val Long.absoluteValue: Long get() = abs(this)

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,\n *   - `0` if the value is zero,\n *   - `1` if the value is positive\n * @see sign function\n */
@SinceKotlin("1.2")
public actual val Long.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}

/**
 * Returns `true` if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Double.isNaN(): Boolean = this != this

/**
 * Returns ‘true’ if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Float.isNaN(): Boolean = this != this

/**
 * Returns \(\text{Double.POSITIVE_INFINITY}\) if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Double.isPositiveInfinity(): Boolean = this == Double.POSITIVE_INFINITY

/**
 * Returns \(\text{Double.NEGATIVE_INFINITY}\) if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Double.isNegativeInfinity(): Boolean = this == Double.NEGATIVE_INFINITY

/**
 * Returns `true` if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Float.isPositiveInfinity(): Boolean = this == Float.POSITIVE_INFINITY

/**
 * Returns `true` if the specified number is an \(\text{NaN}\) value, ‘false’ otherwise.\n * @see NaN\n */
@SinceKotlin("1.2")
public actual fun Float.isNegativeInfinity(): Boolean = this == Float.NEGATIVE_INFINITY

Open Source Used In DNAC Platform Arica 12264
isInfinite() & & isNaN() \n
/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() & & !isNaN()
significant set bit of this [Long] number.

@SinceKotlin("1.3")
public actual fun Long.takeLowestOneBit(): Long =
when (val low = this.low) {
    0 -> Long(0, high.takeLowestOneBit())
else ->
    Long(low.takeLowestOneBit(), 0)
}

* * *

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.takeLowestOneBit(): Long =
    when (val low = this.low) {
        0 -> Long(0, high.takeLowestOneBit())
        else -> Long(low.takeLowestOneBit(), 0)
    }

/**
 * Rotates the binary representation of this [Long] number left by
 * the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter
 * the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as
 * rotating it right by the negated bit count:
 * number.rotateLeft(-n) == number.rotateRight(n)
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateLeft(bitCount: Int): Long {
if ((bitCount and 31) != 0) {
    val low = this.low
    val high = this.high
    val newLow = low shl (bitCount) or high ushr (-bitCount)
    val newHigh = high shl (bitCount) or low ushr (-bitCount)
    return if ((bitCount and 32) == 0) Long(newLow, newHigh) else Long(newHigh, newLow)
} else {
    return if ((bitCount and 32) == 0) this else Long(high, low)
}

/**
 * Rotates the binary representation of this [Long] number right by
 * the specified [bitCount] number of bits.
 * Rotating by a negative bit count is the same as rotating it left by
 * the negated bit count:
 * number.rotateRight(-n) == number.rotateLeft(n)
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public actual inline fun Long.rotateRight(bitCount: Int): Long = rotateLeft(-bitCount)

/**
 * Exposes the JavaScript [Promise object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public open external class Promise<out T>(executor: (resolve: (T) -> Unit, reject: (Throwable) -> Unit) -> Unit) {
    @LowPriorityInOverloadResolution
    public open fun <S> then(onFulfilled: ((T) -> S)?): Promise<S>

    @LowPriorityInOverloadResolution
    public open fun <S> then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>

    public open fun <S> catch(onRejected: (Throwable) -> S): Promise<S>

    companion object {
        public fun <S> all(promise: Promise<in Array<out Promise<in S>>>): Promise<Array<in S>>

        public fun <S> race(promise: Promise<in Array<out Promise<in S>>>): Promise<S>

        public fun reject(e: Throwable): Promise<Nothing>

        public fun <S> resolve(e: Promise<in S>): Promise<in S>
    }
}

// It's workaround for KT-19672 since we can fix it properly until KT-11265 isn't fixed.uninline fun <T, S>
 Promise<T>.then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>
 thisunsafeCast<T>.then(onFulfilled: (T) -> S)?, onRejected: (Throwable) -> S)?:
    Promise<S>

private val INV_2_26: Double = 2.0.pow(-26)
private val INV_2_53: Double = 2.0.pow(-53)
internal actual fun doubleFromParts(hi26: Int, low27: Int): Double =
    hi26 * INV_2_26 + low27 * INV_2_53

/**
 * Represents the constructor of a class. Instances of 'JsClass' can be
 * passed to JavaScript APIs that expect a constructor reference.
 */
external interface JsClass<T : Any> {
}
upperBounds: Array<KType>,\n    variance: String\n) : KTypeParameter {
    val kVariance = when (variance) {
    \"in\" -> KVariance.IN\n    \"out\" -> KVariance.OUT\n    else -> KVariance.INVARIANT\n    }\n    return
    KTypeParameterImpl(name, upperBounds.asList(), kVariance,
    false)\n    val @JsName("getStarKTypeProjection")\n    fun getStarKTypeProjection(): KTypeProjection =\n    KTypeProjection.STAR\n    val @JsName("createCovariantKTypeProjection")\n    fun createCovariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.covariant(type)\n    val @JsName("createInvariantKTypeProjection")\n    fun createInvariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.invariant(type)\n    val @JsName("createContravariantKTypeProjection")\n    fun createContravariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.contravariant(type)\n}

val @JsName("getStarKTypeProjection")\n    fun getStarKTypeProjection(): KTypeProjection =\n    KTypeProjection.STAR\n    val @JsName("createCovariantKTypeProjection")\n    fun createCovariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.covariant(type)\n    val @JsName("createInvariantKTypeProjection")\n    fun createInvariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.invariant(type)\n    val @JsName("createContravariantKTypeProjection")\n    fun createContravariantKTypeProjection(type: KType): KTypeProjection =\n    KTypeProjection.contravariant(type)\n}

internal class DynamicKType : KType {
    override val classifier: KClassifier? = null
    override val arguments: List<KTypeProjection> = emptyList()
    override val isMarkedNullable: Boolean
        = false
    override val annotations: List<Annotation> = emptyList()
    override fun toString(): String = "dynamic"
}

internal fun KVariance.prefixString() =
    when (this) {
        KVariance.INVARIANT -> ""
        KVariance.IN -> "in "
        KVariance.OUT -> "out "
    }
@JsName("floatClass")
val floatClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Float>>(
  \"Float\", \ it is Float )
@JsName("doubleClass")
val doubleClass = PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Double>>(
  \"Double\", \ it is Double )
@JsName("arrayClass")
val arrayClass = PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<Array<*>>>(
  \"Array\", \ it is Array<*> )
@JsName("stringClass")
val stringClass = PrimitiveKClassImpl(js("String"), unsafeCast<JsClass<String>>(
  \"String\", \ it is String )
@JsName("throwableClass")
val throwableClass = PrimitiveKClassImpl(js("Error"), unsafeCast<JsClass<Throwable>>(
  \"Throwable\", \ it is Throwable )
@JsName("booleanArrayClass")
val booleanArrayClass = PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<CharArray>>(
  \"CharArray\")
@JsName("charArrayClass")
val charArrayClass = PrimitiveKClassImpl(js("Uint16Array"), unsafeCast<JsClass<CharArray>>(
  \"CharArray\")
@JsName("byteArrayClass")
val byteArrayClass = PrimitiveKClassImpl(js("Int8Array"), unsafeCast<JsClass<ByteArray>>(
  \"ByteArray\")
@JsName("shortArrayClass")
val shortArrayClass = PrimitiveKClassImpl(js("Int16Array"), unsafeCast<JsClass<ShortArray>>(
  \"ShortArray\")
@JsName("intArrayClass")
val intArrayClass = PrimitiveKClassImpl(js("Int32Array"), unsafeCast<JsClass<IntArray>>(
  \"IntArray\")
@JsName("longArrayClass")
val longArrayClass = PrimitiveKClassImpl(js("Int64Array"), unsafeCast<JsClass<LongArray>>(
  \"LongArray\")
@JsName("floatArrayClass")
val floatArrayClass = PrimitiveKClassImpl(js("Float32Array"), unsafeCast<JsClass<FloatArray>>(
  \"FloatArray\")
@JsName("doubleArrayClass")
val doubleArrayClass = PrimitiveKClassImpl(js("Float64Array"), unsafeCast<JsClass<DoubleArray>>(
  \"DoubleArray\")

@JsName("functionClass")
fun functionClass(arity: Int): KClassImpl<Any> {

  return functionClasses.get(arity) ?: run {
    val result = 
  
  private val functionClasses = 

// a package is omitted to get declarations directly under the module

import kotlin.reflect.*
import kotlin.reflect.js.internal.*

@JsName("getKClass")
internal fun <T : Any>
getKClass(jClass: JsClass<T>): KClass<T> =

@JsName("getKClassFromExpression")
internal fun <T : Any>
getKClassFromExpression(e: T): KClass<T> =
    when (jsTypeOf(e)) {
        "string" ->
        "number" ->
            PrimitiveClasses.intClass
        "boolean" ->
            PrimitiveClasses.booleanClass
        "function" ->
            PrimitiveClasses.functionClass as Dynamic().length
        else ->
            functionClasses.asDynamic()[arity] = 

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)

            when (e)
private fun <T : Any> getOrCreateKClass(jClass: JsClass<T>): KClass<T> {
    if (jClass === js("String")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()

    val metadata = jClass.asDynamic().`$metadata$`

    return if (metadata != null) {
        if (metadata.`$kClass$` == null) {
            val kClass = SimpleKClassImpl(jClass)
            metadata.`$kClass$` = kClass
            kClass
        } else {
            metadata.`$kClass$`
        }
    } else {
        SimpleKClassImpl(jClass)
    }
}

constructor

unsafeCast<KClass<T>>(jClass: JsClass<T>):

getOrCreateKClass(jClass: JsClass<T>):

 KCClass<T> {
    if (jClass === js("String")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()
    val metadata = jClass.asDynamic().`$metadata$`
    return if (metadata != null) {
        if (metadata.`$kClass$` == null) {
            val kClass = SimpleKClassImpl(jClass)
            metadata.`$kClass$` = kClass
            kClass
        } else {
            metadata.`$kClass$`
        }
    } else {
        SimpleKClassImpl(jClass)
    }


String.toBoolean(): Boolean = toLowerCase() == "true"

/**
 * Parses the string as a [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByteArray(): ByteArray = toByteArrayOrNull() ?: numberFormatException(this)

/**
 * Parses the string as a [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatException(this)

private fun String.isNaN(): Boolean = when (this.toLowerCase()) {
    "nan", "+nan", "-nan" -> true
    else -> false
}

/**
 * Checks whether the given [radix] is valid radix for string to number conversion.
 */
@PublishedApi
internal actual fun checkRadix(radix: Int): Int {
    if (radix !in 2..36) {
        throw IllegalArgumentException("radix $radix was not in valid range 2..36")
    }
    return radix
}

internal actual fun digitOf(char: Char, radix: Int): Int = when {
    char >= '0' && char <=
**Public actual inline fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String**

```kotlin
    /**
     * Replaces all occurrences of this regular expression in the specified [input] string with
     * the result of
     * the given function [transform] that takes [MatchResult] and returns a string to be used as a
     * replacement for that match.
     */
    public actual inline fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String {
        var match = find(input)
        if (match == null) return input.toString()

        var lastStart = 0
        val length = input.length
        val sb = StringBuilder(length)

        do {
            val foundMatch = match!!
            sb.append(input, lastStart, foundMatch.range.start)
            sb.append(transform(foundMatch))
            lastStart = foundMatch.range.endInclusive + 1
            match = foundMatch.next()
        } while (lastStart < length && match != null)

        if (lastStart < length) {
            sb.append(input, lastStart, length)
        }

        return sb.toString()
    }
```

**Public actual fun replaceFirst(input: CharSequence, replacement: String): String**

```kotlin
    /**
     * Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression.
     * @param replacement A replacement expression that can include substitutions. See
     */
    public actual fun replaceFirst(input: CharSequence, replacement: String): String {
        val nonGlobalOptions = options.map { it.value }.joinToString(separator = "")
        return input.toString().nativeReplace(RegExp(pattern, nonGlobalOptions), replacement)
    }
```

**Public actual fun split(input: CharSequence, limit: Int = 0): List<String>**

```kotlin
    /**
     * Splits the [input] CharSequence around matches of this regular expression.
     * @param limit Non-negative value specifying the maximum number of substrings
     * the string can be split to.
     * Zero by default means no limit is set.
     */
    actual fun split(input: CharSequence, limit: Int = 0): List<String> {
        require(limit >= 0) { "Limit must be non-negative, but was $limit" }
        val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }
        val result = mutableListOf<String>()
        var lastStart = 0

        for (match in matches) {
            result.add(input.subSequence(lastStart, match.range.start).toString())
            lastStart = match.range.endInclusive + 1
        }
        result.add(input.subSequence(lastStart, input.length).toString())
        return result
    }
```

**Public actual fun toString(): String**

```kotlin
    actual companion object {
        /**
         * Returns a regular expression that
         * matches the specified [literal] string literally.
         * No characters of that string will have special meaning when
         * searching for an occurrence of the regular expression.
         * @param literal A string literal to be matched.
         */
        public actual fun escape(literal: String): String = literal.nativeReplace(patternEscape, "\\$&")
    }
```

**Public actual fun Regex_0(pattern: String, option: RegexOption): Regex**

```kotlin
    /** Creates a regular expression from the specified
     * [pattern] string and the specified single [option].  */
    @Deprecated("Provided for binary compatibility", level = DeprecationLevel.HIDDEN)
    @JsName("Regex_sb3q2\$")
    public fun Regex_0(pattern: String, option: RegexOption): Regex = Regex(pattern, setOf(option))
```

**Public actual fun Regex_1(pattern: String): Regex**

```kotlin
    /** Creates a regular expression from the specified
     * [pattern] string and the default options.  */
    @Deprecated("Provided for binary compatibility", level = DeprecationLevel.HIDDEN)
    @JsName("Regex_61zpoe\$")
    public fun Regex_1(pattern: String): Regex = Regex(pattern, emptySet())
```

**Private fun RegExp.findNext(input: String, from: Int): MatchResult?**

```kotlin
    private fun RegExp.findNext(input: String, from: Int): MatchResult? {
        this.lastIndex = from
        val match = exec(input)
        if (match == null) return null
        val range = match.index..lastIndex - 1

        return object : MatchResult {
            override val range: IntRange = range
            override val value: String
                get() = match[0]!!

            override val groups: MatchGroupCollection = object :
```

---

**Open Source Used In DNAC Platform Arica** 12273
override fun get(index: Int): MatchString = match[index] ?: ""

private fun findNext(input: CharArray, start: Int, range: MatchRange): MatchResult? = 
    when {
        range.isEmpty() -> input.get(start).let { 
            findNext(input, start + 1, range) + 1
        } 
        else -> range.endInclusive + 1
    }
public actual fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) "gi" else "g"),
    Regex.escapeReplacement(newValue))

public actual fun String.replaceFirst(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
    nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "i" else ""), newChar.toString())

public actual interface Appendable {
    public actual fun append(csq: CharSequence?): Appendable
    public actual fun append(csq: CharSequence?, start: Int, end: Int): Appendable
    public actual fun append(c: Char): Appendable
}

public actual class StringBuilder(content: String = "") : Appendable, CharSequence {
    actual constructor(capacity: Int) : this() {}

    actual constructor(content: CharSequence) : this(content.toString()) {}

    actual constructor() : this("")

    private var string: String = content

    actual override val length: Int
        get() = string.asDynamic().length

    actual override fun get(index: Int): Char =
        string.getOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }

    actual override fun subSequence(startIndex: Int, endIndex: Int): CharSequence =
        string.substring(startIndex, endIndex)

    actual override fun append(c: Char): StringBuilder {
        string += c
        return this
    }

    actual override fun append(csq: CharSequence?): StringBuilder {
        string += csq.toString()
        return this
    }

    actual override fun append(csq: CharSequence?, start: Int, end: Int): StringBuilder {
        string += csq.toString().substring(start, end)
        return this
    }

    actual fun append(obj: Any?): StringBuilder {
        string += obj.toString()
        return this
    }

    actual fun reverse(): StringBuilder {
        string = string.asDynamic().split(" ").reverse().join(" ")
        return this
    }

    /**
     * Clears the content of this string builder making it empty.
     * @sample samples.text.Strings.clearStringBuilder
     */
    @SinceKotlin("1.3")
    public fun clear(): StringBuilder {
        string = ""
        return this
    }

    override fun toString(): String = string
}

/**
 * The exception thrown when a character encoding or decoding error occurs.
 * @sample code samples.text.Strings.clearStringBuilder
 */
@SinceKotlin("1.3")
@UseExperimental(ExperimentalStdlibApi::class)
private fun malformed(size: Int, index: Int, throwOnMalformed: Boolean): Int {
    if (throwOnMalformed) throw CharacterCodingException("Malformed sequence starting at ${index - 1}")
    return -size
}

/**
 * Returns code point corresponding to UTF-16 surrogate pair,
 * where the first of the pair is the [high] and the second is in the [string] at the [index].
 * Returns zero if the pair is malformed and [throwOnMalformed] is false.
 * @throws CharacterCodingException if the pair is malformed and [throwOnMalformed] is true.
 * @sample code samples.text.Strings.clearStringBuilder
 */
@SinceKotlin("1.3")
@UseExperimental(ExperimentalStdlibApi::class)
private fun codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (high !in 0xD800..0xDBFF || index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val low = string[index].toInt()
    if
private fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (byte1 and 0x1E == 0 || index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val byte2 = bytes[index].toInt()
    if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }
    return (byte1 shl 6) xor byte2 xor 0xF80
}

private fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF == 0x4) {
        if (byte2 and 0xF0 != 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 12) xor (byte2 shl 6) xor byte3 xor -0x1E080
}

    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0x0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF == 0x4) {
        if (byte2 and 0xF0 != 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0x381F80
}

/**
 * Returns code point corresponding to UTF-8 sequence of
 * two bytes,\n * where the first byte of the sequence is the [byte1] and the second byte is in the [bytes] array at the
 * [index].\n * Returns zero if the sequence is malformed and [throwOnMalformed] is false.\n * @throws CharacterCodingException if the sequence of two bytes is malformed and [throwOnMalformed] is true.\n */
private fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (byte1 and 0x1E == 0 || index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val byte2 = bytes[index].toInt()
    if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }
    return (byte1 shl 6) xor byte2 xor 0xF80
}

/**
 * Returns code point corresponding to UTF-8 sequence of three bytes,\n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].\n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\n * if the sequence is malformed and [throwOnMalformed] is false.\n * @throws CharacterCodingException if the sequence of three bytes is malformed and [throwOnMalformed] is true.\n */
private fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF == 0x4) {
        if (byte2 and 0xF0 != 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 12) xor (byte2 shl 6) xor byte3 xor -0x1E080
}

/**
 * Returns code point corresponding to UTF-8 sequence of four bytes,\n * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].\n * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence\n * if the sequence is malformed and [throwOnMalformed] is false.\n * @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true.\n */
    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0x0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF == 0x4) {
        if (byte2 and 0xF0 != 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0x381F80
}
public class DNAC {
    
    // Escape characters that are not valid for UTF-8 encoding.
    // return char as a byte array.
    @param endIndex the end offset (exclusive) of the substring to encode.
    @param throwOnMalformed whether to throw on malformed char sequence or replace by the
    [REPLACEMENT_BYTE_SEQUENCE].
    
    * @param endIndex the start index of the char sequence to encode.
    * @param throwOnMalformed: Boolean. Encode the char sequence from startIndex to
    endIndex (inclusive) using the encoding specified in the encoding argument.
    * @param encoding: String. The character encoding to use for encoding.
    * @return: The encoded byte array.
    * 
    * @throws CharacterCodingException if the char sequence is
    malformed and [throwOnMalformed] is true. */
    
    public fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {
        require(startIndex >= 0 && endIndex <= string.length &&
        startIndex <= endIndex)
        val bytes = ByteArray(endIndex - startIndex) * MAX_BYTES_PER_CHAR
        var byteIndex = 0
        var charIndex = startIndex
        while (charIndex < endIndex) {
            val code = string[charIndex++].toInt()
            when {
                code < 0x80 ->
                    bytes[byteIndex++] = code.toByte()
                code < 0x800 -> {
                    bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()
                    bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                }
                code < 0xD800 || code >= 0xE000 -> {
                    bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()
                    bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                }
                else -> {
                    val codePoint = codePointFromSurrogate(string, code, charIndex, endIndex, throwOnMalformed)
                    if (codePoint <= 0) {
                        bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[0]
                        bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[1]
                        bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[2]
                    } else {
                        bytes[byteIndex++] = ((codePoint shr 18) or 0xF0).toByte()
                        bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
                        bytes[byteIndex++] = (((codePoint shr 6) and 0x3F) or 0x80).toByte()
                        byteIndex += 3
                    }
                }
            }
        }
        return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
    }

    /**
     * The character a malformed UTF-8 byte sequence is replaced by.
     */
    internal const val REPLACEMENT_CHAR = '\uFFFD'

    /**
     * Decodes the UTF-8 [bytes] array and returns the resulting [String].
     */
    internal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {
        require(startIndex >= 0 && endIndex <= bytes.size && startIndex <= endIndex)
        var byteIndex = startIndex
        val stringBuilder = StringBuilder()

        while (byteIndex < endIndex) {
            val byte = bytes[byteIndex++].toInt()
            when {
                byte >= 0 ->
                    stringBuilder.append(byte.toChar())
                byte shr 5 == -2 -> {
                    val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                    if (code <= 0) {
                        stringBuilder.append(REPLACEMENT_CHAR.toChar())
                        byteIndex += 1
                    } else {
                        byteIndex ++ = -code
                        stringBuilder.append(REPLACEMENT_CHAR.toChar())
                    }
                }
                byte shr 4 == -2 - > {
                    val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                    if (code <= 0) {
                        stringBuilder.append(REPLACEMENT_CHAR.toChar())
                        byteIndex += -code
                    } else {
                        byteIndex ++ = 2
                    }
                }
                byte shr 3 == -2 -> {
                    val code = codePointFrom4(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                    if (code <= 0) {
                        stringBuilder.append(REPLACEMENT_CHAR.toChar())
                        byteIndex += -code
                    } else {
                        byteIndex ++ = 3
                    }
                }
                byteIndex + = -code
            }
        }
        return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
    }

    /*
     * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
     * Use of this source code is governed by the Apache 2.0 license that can be found
     * in the license/LICENSE.txt file.
     */

    package kotlin.time

    @SinceKotlin("1.3")
    @ExperimentalTime
    public actual enum class DurationUnit(internal val scale: Double) {
        /**
         * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
         */
        
    }
}
NANOSECONDS(1e0),

MICROSECONDS(1e3),

MILLISECONDS(1e6),

SECONDS(1e9),

HOURS(3600e9),

DAYS(86400e9),

/**
 * Time unit representing one microsecond, which is 1/1000 of a millisecond.
 */

/**
 * Time unit representing one minute.
 */

/**
 * Time unit representing one hour.
 */

/**
 * Time unit representing one day, which is always equal to 24 hours.
 */

convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {
    val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)
    return when {
        sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)
        sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale)
        else -> value
    }
}
Open Source Used In DNAC Platform Arica 12279

/*

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! */

package org.khronos.webgl

import kotlin.js.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface WebGLContextAttributes {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var depth: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var stencil: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var antialias: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var premultipliedAlpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var preserveDrawingBuffer: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var preferLowPowerToHighPerformance: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var failIfMajorPerformanceCaveat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["alpha"] = alpha
    o["depth"] = depth
    o["stencil"] = stencil
    o["antialias"] = antialias
    o["premultipliedAlpha"] = premultipliedAlpha
    o["preserveDrawingBuffer"] = preserveDrawingBuffer
    o["preferLowPowerToHighPerformance"] = preferLowPowerToHighPerformance
    o["failIfMajorPerformanceCaveat"] = failIfMajorPerformanceCaveat
    return o
}

public external abstract class WebGLObject

/**
 * Exposes the JavaScript [WebGLBuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer) to Kotlin
 */
public external abstract class WebGLBuffer : WebGLObject

/**
 * Exposes the JavaScript [WebGLFramebuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer) to Kotlin
 */
public external abstract class WebGLFramebuffer : WebGLObject

/**
 * Exposes the JavaScript [WebGLProgram](https://developer.mozilla.org/en/docs/Web/API/WebGLProgram) to Kotlin
 */
public external abstract class WebGLProgram : WebGLObject

/**
 * Exposes the JavaScript [WebGLRenderbuffer](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderbuffer) to Kotlin
 */
public external abstract class WebGLRenderbuffer : WebGLObject

/**
 * Exposes the JavaScript [WebGLShader](https://developer.mozilla.org/en/docs/Web/API/WebGLShader) to Kotlin
 */
public external abstract class WebGLShader : WebGLObject

/**
 * Exposes the JavaScript [WebGLTexture](https://developer.mozilla.org/en/docs/Web/API/WebGLTexture) to Kotlin
 */
public external abstract class WebGLTexture : WebGLObject

/**
 * Exposes the JavaScript [WebGLUniformLocation](https://developer.mozilla.org/en/docs/Web/API/WebGLUniformLocation) to Kotlin
 */
public external abstract class WebGLUniformLocation

/**
 * Exposes the JavaScript [WebGLActiveInfo](https://developer.mozilla.org/en/docs/Web/API/WebGLActiveInfo) to Kotlin
 */
public external abstract class WebGLActiveInfo {
    open val size: Int
    open val type: Int
    open val name: String
}

@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WebGLRenderingContextBase {
    val canvas: HTMLCanvasElement
    val drawingBufferWidth: Int
    val drawingBufferHeight: Int
    fun getContextAttributes(): WebGLContextAttributes
    fun isContextLost():
fun pixelStorei(pname: Int, param: Int)
fun polygonOffset(factor: Float, units: Float)
fun renderbufferStorage(target: Int, internalformat: Int)
fun sampleCoverage(value: Float, invert: Boolean)
fun scissor(x: Int, y: Int, width: Int, height: Int)
fun shaderSource(shader: WebGLShader?, source: String)
fun stencilFunc(func: Int, ref: Int, mask: Int)
fun stencilFuncSeparate(face: Int, func: Int, ref: Int, mask: Int)
fun stencilMask(mask: Int)
fun stencilMaskSeparate(face: Int, mask: Int)
fun textureStorage(target: Int, internalformat: Int)
fun texParameterf(target: Int, pname: Int, param: Float)
fun texParameteri(target: Int, pname: Int, param: Int)
fun uniform1f(location: WebGLUniformLocation?, x: Float)
fun uniform1fv(location: WebGLUniformLocation?, x: Float)
fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform1i(location: WebGLUniformLocation?, x: Int)
fun uniform1iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform2f(location: WebGLUniformLocation?, x: Float, y: Float)
fun uniform2fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform2i(location: WebGLUniformLocation?, x: Int, y: Int)
fun uniform2iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform3f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float)
fun uniform3fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform3i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int)
fun uniform3iv(location: WebGLUniformLocation?, v: Int32Array)
fun uniform4f(location: WebGLUniformLocation?, x: Float, y: Float, z: Float, w: Float)
fun uniform4fv(location: WebGLUniformLocation?, v: Float32Array)
fun uniform4i(location: WebGLUniformLocation?, x: Int, y: Int, z: Int, w: Int)
fun uniform4iv(location: WebGLUniformLocation?, v: Int32Array)
fun viewport(x: Int, y: Int, width: Int, height: Int)

companion object {
    val DEPTH_BUFFER_BIT: Int
    val STENCIL_BUFFER_BIT: Int
    val COLOR_BUFFER_BIT: Int
    val POINTS: Int
    val LINES: Int
    val LINE_LOOP: Int
    val LINE_STRIP: Int
    val TRIANGLES: Int
    val TRAPEZOID: Int
    val TRIANGLE_FAN: Int
    val ZERO: Int
    val ONE: Int
    val SRC_COLOR: Int
    val ONE_MINUS_SRC_COLOR: Int
    val SRC_ALPHA: Int
    val ONE_MINUS_SRC_ALPHA: Int
    val DST_COLOR: Int
    val ONE_MINUS_DST_COLOR: Int
    val SRC_ALPHA_SATURATE: Int
val DECR: Int
val INVERT: Int
val INCR_WRAP: Int
val DECR_WRAP: Int
val VENDOR: Int
val RENDERER: Int
val VERSION: Int
val NEAREST: Int
val LINEAR: Int
val NEAREST_MIPMAP_NEAREST: Int
val LINEAR_MIPMAP_NEAREST: Int
val NEAREST_MIPMAP_LINEAR: Int
val LINEAR_MIPMAP_LINEAR: Int
val TEXTURE_MAG_FILTER: Int
val TEXTURE_MIN_FILTER: Int
val TEXTURE_WRAP_S: Int
val TEXTURE_WRAP_T: Int
val TEXTURE_2D: Int
val TEXTURE: Int
val TEXTURE_CUBE_MAP: Int
val TEXTURE_BINDING_CUBE_MAP: Int
val TEXTURE_CUBE_MAP_POSITIVE_X: Int
val TEXTURE_CUBE_MAP_NEGATIVE_X: Int
val TEXTURE_CUBE_MAP_POSITIVE_Y: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
val TEXTURE_CUBE_MAP_POSITIVE_Z: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int
val MAX_CUBE_MAP_TEXTURE_SIZE: Int
val TEXTURE0: Int
val TEXTURE1: Int
val TEXTURE2: Int
val TEXTURE3: Int
val TEXTURE4: Int
val TEXTURE5: Int
val TEXTURE6: Int
val TEXTURE7: Int
val TEXTURE8: Int
val TEXTURE9: Int
val TEXTURE10: Int
val TEXTURE11: Int
val TEXTURE12: Int
val TEXTURE13: Int
val TEXTURE14: Int
val TEXTURE15: Int
val TEXTURE16: Int
val TEXTURE17: Int
val TEXTURE18: Int
val TEXTURE19: Int
val TEXTURE20: Int
val TEXTURE21: Int
val TEXTURE22: Int
val TEXTURE23: Int
val TEXTURE24: Int
val TEXTURE25: Int
val TEXTURE26: Int
val TEXTURE27: Int
val TEXTURE28: Int
val TEXTURE29: Int
val TEXTURE30: Int
val TEXTURE31: Int
val ACTIVE_TEXTURE: Int
val REPEAT: Int
val CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int
val FLOAT_VEC3: Int
val FLOAT_VEC4: Int
val INT_VEC2: Int
val INT_VEC3: Int
val INT_VEC4: Int
val BOOL: Int
val BOOLEAN_VEC2: Int
val BOOLEAN_VEC3: Int
val BOOLEAN_VEC4: Int
val FLOAT_MAT2: Int
val FLOAT_MAT3: Int
val FLOAT_MAT4: Int
val SAMPLER_2D: Int
val SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_STRIDE: Int
val VERTEX_ATTRIB_ARRAY_TYPE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val COMPILATION_STATUS: Int
val LOW_FLOAT: Int
val MEDIUM_FLOAT: Int
val HIGH_FLOAT: Int
val LOW_INT: Int
val MEDIUM_INT: Int
val HIGH_INT: Int
val FRAMEBUFFER: Int
val RENDERBUFFER: Int
val RGBA4: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT16: Int
val STENCIL_INDEX: Int
val STENCIL_INDEX8: Int
val DEPTH_STENCIL: Int
val RENDERBUFFER_WIDTH: Int
val RENDERBUFFER_HEIGHT: Int
val RENDERBUFFER_INTERNAL_FORMAT: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_BLUE_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val RENDERBUFFER: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_BLUE_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val MAX_RENDERBUFFER_SIZE: Int
val RENDERBUFFER_BINDING: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
UNPACK_FLIP_Y_WEBGL: Int
UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
CONTEXT_LOST_WEBGL: Int
UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
BROWSER_DEFAULT_WEBGL: Int
} * Exposes the JavaScript [WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to Kotlin

*Unpublic external abstract class WebGLRenderingContext : WebGLRenderingContextBase, RenderingContext

* companion object {
    val DEPTH_BUFFER_BIT: Int
    val STENCIL_BUFFER_BIT: Int
    val POINTS: Int
    val LINES: Int
    val LINE_LOOP: Int
    val TRIANGLES: Int
    val TRIANGLE_STRIP: Int
    val TRIANGLE_FAN: Int
    val ZERO: Int
    val ONE: Int
    val SRC_COLOR: Int
    val ONE_MINUS_SRC_COLOR: Int
    val SRC_ALPHA: Int
    val ONE_MINUS_SRC_ALPHA: Int
    val DST_ALPHA: Int
    val ONE_MINUS_DST_ALPHA: Int
    val DST_COLOR: Int
    val ONE_MINUS_DST_COLOR: Int
    val SRC_ALPHA_SATURATE: Int
    val BLEND_EQUATION: Int
    val BLEND_EQUATION_RGB: Int
    val BLEND_EQUATION_ALPHA: Int
    val FUNC_ADD: Int
    val BLEND_SRC_RGB: Int
    val BLEND_SRC_ALPHA: Int
    val BLEND_DST_RGB: Int
    val BLEND_DST_ALPHA: Int
    val BLEND_COLOR: Int
    val ARRAY_BUFFER: Int
    val ELEMENT_ARRAY_BUFFER: Int
    val ARRAY_BUFFER_BINDING: Int
    val FRONT: Int
    val BACK: Int
    val CULL_FACE: Int
    val BLEND: Int
    val DITHER: Int
    val STENCIL_TEST: Int
    val DEPTH_TEST: Int
    val SCISSOR_TEST: Int
    val POLYGON_OFFSET_FILL: Int
    val SAMPLE_ALPHA_TO_COVERAGE: Int
    val SAMPLE_COVERAGE: Int
    val NO_ERROR: Int
    val INVALID_ENUM: Int
    val INVALID_VALUE: Int
    val INVALID_OPERATION: Int
    val OUT_OF_MEMORY: Int
    val CW: Int
    val CCW: Int
    val LINE_WIDTH: Int
    val ALIASED_POINT_SIZE_RANGE: Int
    val ALIASED_LINE_WIDTH_RANGE: Int
    val Red_BITS: Int
    val Green_BITS: Int
    val Blue_BITS: Int
    val Alpha_BITS: Int
    val Depth_BITS: Int
    val Stencil_BITS: Int
    val POLYGON_OFFSET_UNITS: Int
    val POLYGON_OFFSET_FACTOR: Int
    val TEXTURE_BINDING_2D: Int
    val SAMPLE_BUFFERS: Int
    val SAMPLES: Int
    val SAMPLE_COVERAGE_VALUE: Int
    val SAMPLE_COVERAGE_INVERT: Int
    val COMPRESSED_TEXTURE_FORMATS: Int
    val DONT_CARE: Int
    val FASTEST: Int
    val NICEST: Int
    val GENERATE_MIPMAP_HINT: Int
    val BYTE: Int
    val UNSIGNED_BYTE: Int
    val SHORT: Int
    val UNSIGNED_SHORT: Int
    val INT: Int
    val UNSIGNED_INT: Int
    val FLOAT: Int
    val DEPTH_COMPONENT: Int
    val RGBA: Int
    val LUMINANCE: Int
    val LUMINANCE_ALPHA: Int
RENDERBUFFER_STENCIL_SIZE: Int
        val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
        val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
        val COLOR_ATTACHMENT0: Int
        val DEPTH_ATTACHMENT: Int
        val STENCIL_ATTACHMENT: Int
        val DEPTH_STENCIL_ATTACHMENT: Int
        val NONE: Int
        val FRAMEBUFFER_COMPLETE: Int
        val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
        val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
        val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
        val FRAMEBUFFER_UNSUPPORTED: Int
        val FRAMEBUFFER_BINDING: Int
        val RENDERBUFFER_BINDING: Int
        val MAX_RENDERBUFFER_SIZE: Int
        val INVALID_FRAMEBUFFER_OPERATION: Int
        val UNPACK_FLIP_Y_WEBGL: Int
        val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
        val CONTEXT_LOST_WEBGL: Int
        val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
        val BROWSER_DEFAULT_WEBGL: Int
    }
}

/**
 * Exposes the JavaScript
 to Kotlin

*/
public external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit =
definedExternally): Event { open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WebGLContextEventInit : EventInit {
    var statusMessage: String? /* = "" */

    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun WebGLContextEventInit(statusMessage: String? = "", bubbles: Boolean? = false,
WebGLContextEventInit {
    var o = js("({})")
    o["statusMessage"] = statusMessage
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer)
 to Kotlin

*/
public external open class ArrayBuffer(length: Int) : BufferDataSource {
    open val byteLength: Int
    fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer

    companion object {
        fun isView(value: Any?): Boolean
    }
}

/**
 * Exposes the JavaScript
[ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView)
 to Kotlin

*/
public external interface ArrayBufferView : BufferDataSource {
    val buffer: ArrayBuffer
    val byteOffset: Int
    val byteLength: Int
    fun set(array: ArrayBufferView, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): ArrayBufferView

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun ArrayBufferView.get(index: Int): Byte = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun ArrayBufferView.set(index: Int, value: Byte) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript
[Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array)
 to Kotlin

*/
public external open class Int8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Int8Array)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int =
definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int8Array, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int8Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Int8Array.get(index: Int): Byte = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Int8Array.set(index: Int, value: Byte) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript
[Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array)
 to Kotlin

*/
public external open class Uint8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Uint8Array)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int =
definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Uint8Array, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint8Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Uint8Array.get(index: Int): Byte = asDynamic()[index]
fun Uint8Array.set(index: Int, value: Byte) { asDynamic()[(index) = value } /* Exposes the JavaScript Uint8ClampedArray [https://developer.mozilla.org/en/docs/Web/API/Uint8ClampedArray] to Kotlin */

external open class Uint8ClampedArray : ArrayBufferView (\n constructor(length: Int)\n constructor(array: Array<Byte>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n fun set(array: Uint8ClampedArray, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Uint8ClampedArray\n companion object {\n BYTES_PER_ELEMENT: Int\n }\n @kotlin.internal.InlineOnly\n open inline fun operator fun Uint8ClampedArray.get(index: Int): Byte = asDynamic()[(index) \n \n \n\n Kotlin */

external open class Uint16Array : ArrayBufferView (\n constructor(length: Int)\n constructor(array: Array<Short>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n fun set(array: Uint16Array, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Uint16Array\n companion object {\n BYTES_PER_ELEMENT: Int\n }\n @kotlin.internal.InlineOnly\n open inline fun operator fun Uint16Array.get(index: Int): Short = asDynamic()[(index) \n \n\n Kotlin */

external open class Uint32Array : ArrayBufferView (\n constructor(length: Int)\n constructor(array: Array<Int>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n fun set(array: Uint32Array, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Uint32Array\n companion object {\n BYTES_PER_ELEMENT: Int\n }\n @kotlin.internal.InlineOnly\n open inline fun operator fun Uint32Array.get(index: Int): Int = asDynamic()[(index) \n \n\n Kotlin */

external open class Float32Array : ArrayBufferView (\n constructor(length: Int)\n constructor(array: Array<Float>)\n constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)\n open val length: Int\n override val buffer: ArrayBuffer\n override val byteOffset: Int\n fun set(array: Float32Array, offset: Int = definedExternally)\n fun subarray(start: Int, end: Int): Float32Array\n companion object {\n BYTES_PER_ELEMENT: Int\n }\n @kotlin.internal.InlineOnly\n open inline fun operator fun Float32Array.get(index: Int): Float = asDynamic()[(index) \n \n\n Kotlin */

Open Source Used In DNAC Platform Arica 12287
class Float32Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float32Array)
    constructor(array: Array<Float>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float32Array, offset: Int = definedExternally)
    fun set(array: Array<Float>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.get(index: Int): Float = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.set(index: Int, value: Float) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin
 */
public external open class Float64Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float64Array)
    constructor(array: Array<Double>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float64Array, offset: Int = definedExternally)
    fun set(array: Array<Double>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float64Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.get(index: Int): Double = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin
 */
public external open class DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) : ArrayBufferView {
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun getInt8(byteOffset: Int): Byte
    fun getUint8(byteOffset: Int): Byte
    fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
    fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double
    fun setInt8(byteOffset: Int, value: Byte)
    fun setUint8(byteOffset: Int, value: Byte)
    fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
    fun setUint16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
    fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally)
    fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally)
}

public external interface BufferDataSource

public external interface TexImageSource
close class ClipboardEvent(type: String, eventoInitDict: ClipboardEventInit = definedExternally) : Event {
    open val clipboardData: DataTransfer?
    val companion object {

        fun readText(): Promise<String>

    }
}

/**
 * Exposes the JavaScript [Clipboard](https://developer.mozilla.org/en/docs/Web/API/Clipboard) to Kotlin
 */
public external abstract class Clipboard : EventTarget {
    fun read(): Promise<DataTransfer>
    fun readText(): Promise<String>
    fun write(data: DataTransfer): Promise<Unit>
    fun writeText(data: String): Promise<Unit>
}

public external interface ClipboardPermissionDescriptor {

    var allowWithoutGesture: Boolean? /* = false */
                get() = definedExternally
                set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline operator fun ClipboardPermissionDescriptor(allowWithoutGesture: Boolean? = false): ClipboardPermissionDescriptor {
    val o = js("({})")
    o["allowWithoutGesture"] = allowWithoutGesture
    return o
}

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details

package org.w3c.dom.css

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external abstract class MediaList : ItemArrayLike<String> {
    open var mediaText: String
    fun appendMedium(medium: String)
    fun deleteMedium(medium: String)
    fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun MediaList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript [StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin
 */
public external abstract class StyleSheet {
    open var media: MediaList
    open val type: String
    open var disabled: Boolean
    open val href: String?
    open val ownerNode: UnionElementOrProcessingInstruction?
    open val parentStyleSheet: StyleSheet?
    open val title: String?
}

/**
 * Exposes the JavaScript [CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin
 */
public external abstract class CSSStyleSheet : StyleSheet {
    open var cssRules: CSSRuleList
    open var ownerRule: CSSRule
    open val ownerStyleSheet: StyleSheet
}

/**
 * Exposes the JavaScript [StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin
 */
public external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {
    override fun item(index: Int): StyleSheet?
}

@kotlin.internal.InlineOnly
public inline operator fun StyleSheetList.get(index: Int): StyleSheet? = asDynamic()[index]
CSSStyleDeclaration companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}

unpublic external abstract class CSSImportRule : CSSRule {
val href: String
val media: MediaQueryList
val styleSheet: CSSStyleSheet
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

public external abstract class CSSGroupingRule : CSSRule {
open val cssRules: CSSRuleList
fun insertRule(rule: String, index: Int): Int
fun deleteRule(index: Int)
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

/**
 * Exposes the JavaScript [CSSGroupingRule](https://developer.mozilla.org/en/docs/Web/API/CSSGroupingRule) to Kotlin
 */
public external abstract class CSSMediaRule : CSSGroupingRule {
open val media: MediaQueryList
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

/**
 * Exposes the JavaScript [CSSPageRule](https://developer.mozilla.org/en/docs/Web/API/CSSPageRule) to Kotlin
 */
public external abstract class CSSPageRule : CSSGroupingRule {
open var selectorText: String
open val style: CSSStyleDeclaration
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

public external abstract class CSSMarginRule : CSSRule {
open val name: String
open val style: CSSStyleDeclaration
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

/**
 * Exposes the JavaScript [CSSNamespaceRule](https://developer.mozilla.org/en/docs/Web/API/CSSNamespaceRule) to Kotlin
 */
public external abstract class CSSNamespaceRule : CSSRule {
open val namespaceURI: String
open val prefix: String
companion object {
val STYLE_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}
}

/**
 * Exposes the JavaScript [CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin
 */
public external abstract class CSSStyleDeclaration : ItemArrayLike<String> {
open var cssText: String
open var parentRule: CSSRule?
open var cssFloat: String
open var alignContent: String
open var alignItems: String
open var alignSelf: String
open var animation: String
open var animationDelay: String
open var animationDirection: String
open var animationDuration: String
open var animationFillMode: String
open var animationIterationCount: String
open var animationName: String
open var animationPlayState: String
open var animationTimingFunction: String
open var backfaceVisibility: String
open var background: String
open var backgroundAttachment: String
open var backgroundColor: String
open var backgroundImage: String
open var backgroundOrigin: String
open var backgroundPosition: String
open var backgroundRepeat: String
open var backgroundSize: String
open var border: String
open var borderBottom: String
open var borderBottomColor: String
open var borderBottomLeftRadius: String
open var borderBottomRightRadius: String
open var borderBottomStyle: String
open var borderBottomWidth: String
open var borderCollapse: String
open var borderColor:
String
  open var textJustify: String
  open var textOrientation: String
  open var textOverflow: String
  open var top: String
  open var transform: String
  open var transitionDelay: String
  open var transitionDuration: String
  open var transitionProperty: String
  open var transitionTimingFunction: String
  open var unicodeBidi: String
  open var verticalAlign: String
  open var visibility: String
  open var whiteSpace: String
  open var wordBreak: String
  open var wordSpacing: String
  open var wordWrap: String
  open var zIndex: String

open var _dashed_attribute: String
open var _camel_cased_attribute: String
open var _webkit_cased_attribute: String
fun getPropertyValue(property: String): String
fun getPriority(property: String): String
fun setProperty(property: String, value: String, priority: String = definedExternally)
fun setPropertyValue(property: String, value: String)
fun setPropertyPriority(property: String, priority: String)
fun removeProperty(property: String): String
override fun item(index: Int): String

@kotlin.internal.InlineOnly
public inline operator fun CSSStyleDeclaration.get(index: Int): String? = asDynamic()[index]

public external interface ElementCSSInlineStyle {
    val style: CSSStyleDeclaration
}

/**
 * Exposes the JavaScript
[CSS](https://developer.mozilla.org/en/docs/Web/API/CSS) to Kotlin
 */
public external abstract class CSS {

    companion object {
        fun escape(ident: String): String
    }
}

public external interface UnionElementOrProcessingInstruction

**
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.

 * NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!

 * See
 * github.com/kotlin/dukat for details

package org.w3c.dom.events

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript
[UIEvent](https://developer.mozilla.org/en/docs/Web/API/UIEvent) to Kotlin
 */
public external open class UIEvent(type: String, eventInitDict: UIEventInit = definedExternally) : Event {

    open val view: Window?

    open val detail: Int

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface UIEventInit : EventInit {
        var view: Window? /* = null */
            get() = definedExternally
            set(value) = definedExternally
        var detail: Int? /* = 0 */
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }


    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

composed\n return o
})\n*/
* Exposes the JavaScript [MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin\n*\npublic external open class MouseEvent\n(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent,
UnionElementOrMouseEvent\{\n open val screenX: Int\n open val screenY: Int\n open val clientX: Int\n open val clientY: Int\n open val ctrlKey: Boolean\n open val shiftKey: Boolean\n open val altKey: Boolean\n open val metaKey: Boolean\n open val button: Short\n open val buttons: Short\n open val relatedTarget: EventTarget\n? open val region: String\n open val pageX: Double\n open val pageY: Double\n open val x: Double\n open val y: Double\n open val offsetX: Double\n open val offsetY: Double\n fun getModifierState(keyArg: String): Boolean
}

/**
 * Exposes the JavaScript [MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin
 */
public external open
class MouseEvent\n(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent,
UnionElementOrMouseEvent {
    open val screenX: Int
    open val screenY: Int
    open val clientX: Int
    open val clientY: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean
    open val button: Short
    open val buttons: Short
    open val relatedTarget: EventTarget?
    open val region: String?
    open val pageX: Double
    open val pageY: Double
    open val x: Double
    open val y: Double
    open val offsetX: Double
    open val offsetY: Double
    fun getModifierState(keyArg: String): Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MouseEventInit : EventModifierInit {
    var screenX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var screenY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var clientX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var clientY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var button: Short? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var buttons: Short? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var region: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

Open Source Used in DNAC Platform Arica  12293
/*
get() = definedExternally
set(value) = definedExternally
var modifierSymbolLock: Boolean? /* = false */
get() = definedExternally
set(value) = definedExternally
*/

get() = definedExternally
set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 [WheelEvent](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin
 */
public external open class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {
    open val deltaX: Double
    open val deltaY: Double
    open val deltaZ: Double
    open val deltaMode: Int

    companion object {
        val DOM_DELTA_PIXEL: Int
        val DOM_DELTA_LINE: Int
        val DOM_DELTA_PAGE: Int
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WheelEventInit : MouseEventInit {
    var deltaX: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaY: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaZ: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaMode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["deltaX"] = deltaX
    o["deltaY"] = deltaY
    o["deltaZ"] = deltaZ
    o["deltaMode"] = deltaMode
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 [InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to Kotlin
 */
public external open class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) : UIEvent {
    open val data: String
    open val isComposing: Boolean

    companion object {
        val NONE:
open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {

    open val key: String
    open val code: String
    open val location: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean
    open val repeat: Boolean

    open val isComposing: Boolean
    open val charCode: Int
    open val keyCode: Int
    open val which: Int

    fun getModifierState(keyArg: String): Boolean

    companion object {
        val DOM_KEY_LOCATION_STANDARD: Int
        val DOM_KEY_LOCATION_LEFT: Int
        val DOM_KEY_LOCATION_RIGHT: Int
        val DOM_KEY_LOCATION_NUMPAD: Int
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface KeyboardEventInit : EventModifierInit {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var code: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var location: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var repeat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ctrlKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierAltGraph: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierCapsLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierFn: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierHyper: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierNumLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierScrollLock: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var modifierSuper: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var view: Window? = null
    var detail: Int? = 0
    var bubbles: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

public external interface CompositionEventInit : InputEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var view: Window? = null
    var detail: Int? = 0
    var bubbles: Boolean? /* = false */
        cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

public external class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) : UIEvent {

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface InputEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var view: Window? = null
    var detail: Int? = 0
    var bubbles: Boolean? /* = false */
        cancelable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}
```kotlin
open class Event(type: String, eventInitDict: EventInit = definedExternally) {
    open val type: String
    open val target: EventTarget?
    open val currentTarget: EventTarget?
    open val eventPhase: Short
    open val bubbles: Boolean
    open val cancelable: Boolean
    open val defaultPrevented: Boolean
    open val composed: Boolean
    open val isTrusted: Boolean
    open val timeStamp: Number

    fun composedPath(): Array<EventTarget>
    fun stopPropagation()
    fun stopImmediatePropagation()
    fun preventDefault()
    fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean)
}

fun handleEvent(event: Event)

fun addEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)
fun addEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic = definedExternally)
fun removeEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)
fun removeEventListener(type: String, callback: ((Event) -> Unit)?, options: dynamic = definedExternally)
fun dispatchEvent(event: Event): Boolean
}

fun HTMLAllCollection.prototype.item(nameOrIndex: String = definedExternally): UnionElementOrHTMLCollection?
fun HTMLAllCollection.prototype.namedItem(name: String): UnionElementOrHTMLCollection?

fun HTMLCollection.prototype.item(nameOrIndex: String = definedExternally): Element?
fun HTMLCollection.prototype.namedItem(name: String): Element?

fun HTMLOptionsCollection.prototype.add(element: HTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
fun HTMLOptionsCollection.prototype.remove(index: Int)

fun RadioNodeList.prototype.value: String

fun RadioNodeList.prototype.add(element: HTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
fun RadioNodeList.prototype.remove(index: Int)
```

---

Open Source Used In DNAC Platform Arica 12296
open var tabIndex: Int
    open var accessKey: String
    open var accessKeyLabel: String
    open var draggable: Boolean
    open var dropzone: DOMTokenList
    open var contextMenu: HTMLMenuElement
    open var spellcheck: Boolean
    open var innerText: String
    open val offsetParent: Element?
    open val offsetTop: Int
    open val offsetLeft: Int
    open val offsetWidth: Int
    open val offsetHeight: Int
    fun click()
    fun focus()
    fun blur()
    fun forceSpellCheck()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript [HTMLUnknownElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUnknownElement) to Kotlin
 */
public external abstract class HTMLUnknownElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript [DOMStringMap](https://developer.mozilla.org/en/docs/Web/API/DOMStringMap) to Kotlin
 */
public external abstract class DOMStringMap

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.get(name: String): String? = asDynamic()[name]

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.set(name: String, value: String) { asDynamic()[name] = value }

/**
 * Exposes the JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin
 */
public external abstract class HTMLHtmlElement : HTMLElement {
    open var version: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript [HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin
 */
public external abstract class HTMLHeadElement : HTMLElement
Open Source Used In DNAC Platform Arica 12300
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

**Exposes the JavaScript [HTMLUListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUListElement) to Kotlin**

```kotlin
public external abstract class HTMLUListElement : HTMLElement {
    open var compact: Boolean
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [HTMLLIElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLIElement) to Kotlin**

```kotlin
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [HTMLDListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDListElement) to Kotlin**

```kotlin
public external abstract class HTMLDListElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin**

```kotlin
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

**Exposes the JavaScript [HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin**

```kotlin
public external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {
    open
var target: String
open var download: String
open var ping: String
open var rel: String
open var relList: DOMTokenList
open var hreflang: String
open var type: String
open var text: String
open var referrerPolicy: String
open var coords: String
open var charset: String
open var name: String
open var rev: String
open var shape: String
open val ATTRIBUTE_NODE: Short
open val TEXT_NODE: Short
open val CDATA_SECTION_NODE: Short
open val ENTITY_REFERENCE_NODE: Short
open val ENTITY_NODE: Short
open val PROCESSING_INSTRUCTION_NODE: Short
open val COMMENT_NODE: Short
open val DOCUMENT_NODE: Short
open val DOCUMENT_TYPE_NODE: Short
open val DOCUMENT_FRAGMENT_NODE: Short
open val NOTATION_NODE: Short
open val DOCUMENT_POSITION_DISCONNECTED: Short
open val DOCUMENT_POSITION_PRECEDING: Short
open val DOCUMENT_POSITION_FOLLOWING: Short
open val DOCUMENT_POSITION_CONTAINS: Short
open val DOCUMENT_POSITION_CONTAINED_BY: Short
open val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Exposes the JavaScript [HTMLDataElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataElement) to Kotlin

Exposes the JavaScript [HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin

Exposes the JavaScript [HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin

Exposes the JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin

Open Source Used In DNAC Platform Arica 12302
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 */
public external interface HTMLHyperlinkElementUtils {
    var href: String
    val origin: String
    var protocol: String
    var username: String
    var password: String
    var host: String
    var hostname: String
    var port: String
    var pathname: String
    var search: String
    var hash: String
}

/**
 * Exposes the JavaScript HTMLModElement=https://developer.mozilla.org/en/docs/Web/API/HTMLModElement to Kotlin
 */
public external abstract class HTMLModElement : HTMLElement {
    open var cite: String
    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLPictureElement=https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement to Kotlin
 */
public external abstract class HTMLPictureElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLSourceElement=https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement to Kotlin
 */
public external abstract class HTMLSourceElement : HTMLElement {
    open var src: String
    open var type: String
    open var srcset: String
    open var sizes: String
    open var media: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Open Source Used In DNAC Platform Arica  12304

/*Anpublic external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource |\n  open var alt: String?\n  open var src: String?\n  open var srcset: String?\n  open var crossOrigin: String?\n  open var useMap: String?\n  open var isMap: Boolean\n  open var width: Int?\n  open var height: Int?\n  open val naturalWidth: Int?\n  open val naturalHeight: Int?\n  open val complete: Boolean\n  open val currentSrc: String?\n  open var referrerPolicy: String?\n  open var name: String?\n  open var lowsrc: String?\n  open var align: String?\n  open var hspace: Int?\n  open var vspace: Int?\n  open var longDesc: String?\n  open var border: String?\n  open var x: Int?\n  open var y: Int?\n  companion object {\n    val ELEMENT_NODE: Short\n    val ATTRIBUTE_NODE: Short\n    val TEXT_NODE: Short\n    val CDATA_SECTION_NODE: Short\n    val ENTITY_REFERENCE_NODE: Short\n    val ENTITY_NODE: Short\n    val PROCESSING_INSTRUCTION_NODE: Short\n    val COMMENT_NODE: Short\n    val DOCUMENT_NODE: Short\n    val DOCUMENT_FRAGMENT_NODE: Short\n    val DOCUMENT_POSITION_DISCONNECTED: Short\n    val DOCUMENT_POSITION_PRECEDING: Short\n    val DOCUMENT_POSITION_FOLLOWING: Short\n    val DOCUMENT_POSITION_CONTAINS: Short\n    val DOCUMENT_POSITION_CONTAINED_BY: Short\n    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n  }\n*/

/**
 * Exposes the JavaScript

  [HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin

  */
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String
    open var srcdoc: String
    open var name: String
    open val sandbox: DOMTokenList
    open var allowFullscreen: Boolean
    open var allowUserMedia: Boolean
    open var width: String
    open var height: String
    open var referrerPolicy: String
    open val contentDocument: Document?
    open val contentWindow: Window?
    open var align: String
    open var scrolling: String
    open var frameborder: String
    open var longDesc: String
    open var marginHeight: String
    open var marginWidth: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript

  [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin

  */
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String
    open var type: String
    open var width: String
    open var height: String
    open var align: String
    open var name: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript

  [HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin

  */
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String
    open var
type: String
open var typeMustMatch: Boolean
open var name: String
open var useMap: String
open val form: HTMLFormElement?
open var width: String
open var height: String
open val contentDocument: Document?
open val contentWindow: Window?
open val willValidate: Boolean
open val open var validity: ValidityState?
open val validationMessage: String
open var var code: String
open var var align: String
open var open var declare: Boolean
open var open var hspace: Int
open var standby:
open var open var vspace: Int
open var open var codeBase: String?
open var open var codeType: String
open var open var border:
open var fun getSVGDocument(): Document?
open fun fun checkValidity(): Boolean
open fun fun reportValidity(): Boolean
open fun setCustomValidity(error: String)

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin
*/
public external abstract class HTMLParamElement : HTMLElement {
open var name: String
open var value: String
open var type: String
open var valueType: String

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin
*/
public external abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {
open var width: Int
open var height: Int
open val videoWidth: Int
open val videoHeight: Int
open var poster: String
open var playsInline: Boolean

companion object {
val NETWORK_EMPTY: Short
val NETWORK_IDLE: Short
val NETWORK_LOADING: Short
val NETWORK_NO_SOURCE: Short
val HAVE_NOTHING: Short
val HAVE_METADATA: Short
val HAVE_CURRENT_DATA: Short
val HAVE_FUTURE_DATA: Short
val HAVE_ENOUGH_DATA: Short
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin
*/
public external abstract class HTMLAudioElement : HTMLMediaElement {

companion object {
val NETWORK_EMPTY: Short
val NETWORK_IDLE: Short
val NETWORK_LOADING: Short
val NETWORK_NO_SOURCE: Short
val HAVE_NOTHING: Short
val HAVE_METADATA: Short
val HAVE_CURRENT_DATA: Short
val HAVE_FUTURE_DATA: Short
val HAVE_ENOUGH_DATA: Short
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}*\n*/
// Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin
public external abstract class HTMLTrackElement : HTMLElement {
    open var kind: String
    open var src: String
    open var srclang: String
    open var label: String
    open var default: Boolean
    open val readyState: Short
    open val track: TextTrack

    companion object {
        val NONE: Short
        val LOADING: Short
        val LOADED: Short
        val ERROR: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to Kotlin
 */
public external abstract class HTMLMediaElement : HTMLElement {
    open val error: MediaError?
    open var src: String
    open var srcObject: dynamic
    open val currentSrc: String
    open var crossOrigin: String?
    open val networkState: Short
    open var preload: String
    open val buffered: TimeRanges
    open val readyState: Short
    open var seeking: Boolean
    open var currentTime: Double
    open var duration: Double
    open var paused: Boolean
    open var defaultPlaybackRate: Double
    open var playbackRate: Double
    open var played: TimeRanges
    open var seekable: TimeRanges
    open var ended: Boolean
    open var autoplay: Boolean
    open var loop: Boolean
    open var controls: Boolean
    open var volume: Double
    open var muted: Boolean
    open var defaultMuted: Boolean
    open var audioTracks: AudioTrackList
    open var videoTracks: VideoTrackList
    open val textTracks: TextTrackList
    fun load()
    fun canPlayType(type: String): CanPlayTypeResult
    fun fastSeek(time: Double)
    fun getStartDate(): dynamic
    fun play(): Promise<Unit>
    fun pause()
    fun addTextTrack(kind: TextTrackKind, label: String = definedExternally,
        language: String = definedExternally): TextTrack

    companion object {
        val NETWORK_NO_SOURCE: Short
        val HAVE NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```kotlin
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
val DOCUMENT_POSITION_NODE: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION:absolute: Short
val DOCUMENT_POSITION:relative: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript
 [MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin
 */
public external abstract class MediaError {
    open val code: Short

    companion object {
        val MEDIA_ERR_ABORTED: Short
        val MEDIA_ERR_NETWORK: Short
        val MEDIA_ERR_DECODE: Short
        val MEDIA_ERR_SRC_NOT_SUPPORTED: Short
    }
}

/**
 * Exposes the JavaScript
 [AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin
 */
public external abstract class AudioTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): AudioTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun AudioTrackList.get(index: Int): AudioTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin
 */
public external abstract class VideoTrackList : EventTarget {
    open val length: Int
    open val selectedIndex: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): VideoTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun VideoTrackList.get(index: Int): VideoTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [TextTrackList](https://developer.mozilla.org/en/docs/Web/API/TextTrackList) to Kotlin
 */
public external abstract class TextTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): TextTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackList.get(index: Int): TextTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [TextTrack](https://developer.mozilla.org/en/docs/Web/API/TextTrack) to Kotlin
 */
public external abstract class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val kind: TextTrackKind
    open val label: String
    open val language: String
    open val id: String
    open val inBandMetadataTrackDispatchType: String
    open var mode: TextTrackMode
    open val cues: TextTrackCueList?
    open var oncuechange: ((Event) -> dynamic)?
    fun addCue(cue: TextTrackCue)
    fun removeCue(cue: TextTrackCue)
}

public external abstract class TextTrackCueList {
    open val length: Int
    fun getCueById(id: String): TextTrackCue?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackCueList.get(index: Int): TextTrackCue? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [TimeRanges](https://developer.mozilla.org/en/docs/Web/API/TimeRanges) to Kotlin
 */
public external abstract class TimeRanges {
    open val length: Int
    fun start(index: Int): Double
    fun end(index: Int): Double
}

/**
 * Exposes the JavaScript
 [TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin
 */
public external open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {
    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}
```

Open Source Used In DNAC Platform Arica  12307
UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
        get() = definedExternally
        set(value) =
            definedExternally
}

@kotlin.internal.InlineOnly
public inline fun TrackEventInit(track:
        UnionAudioTrackOrTextTrackOrVideoTrack? = null, bubbles: Boolean? = false, cancelable: Boolean? = false,
        composed: Boolean? = false): TrackEventInit {
        val o = js("({})")
        o["track"] = track
        o["bubbles"] =
            bubbles
        o["Cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

/**
 * Exposes the JavaScript [HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin
 */
public external abstract class HTMLMapElement : HTMLElement {
    open var name: String
    open val
        areas: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin
 */
public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var
        alt: String
        coords: String
        shape: String
        target: String
        download: String
        ping: String
        rel: String
        relList: DOMTokenList
        referrerPolicy: String
        noHref: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var
        caption: HTMLTableCaptionElement?
        tHead: HTMLTableSectionElement?
        tFoot: HTMLTableSectionElement?
        tBodies: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

*Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin

public external abstract class HTMLTableCaptionElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin
 */
public external abstract class HTMLTableColElement : HTMLElement {
    open var span: Int
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin
 */
public external abstract class HTMLTableSectionElement : HTMLElement {
    open val rows: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    fun insertRow(index: Int = definedExternally): HTMLElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin
 */
public external abstract class HTMLTableRowElement : HTMLElement {
    open val rowIndex: Int
    open val sectionRowIndex: Int
    open val cells: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var bgColor: String
    fun insertCell(index: Int = definedExternally): HTMLElement
    fun deleteCell(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin

An external abstract class HTMLTableCellElement: HTMLElement { open var colSpan: Int; open var rowSpan: Int; open var cellIndex: Int; open var scope: String; open var abbr: String; open var align: String; open var axis: String; open var width: String; open var ch: String; open var chOff: String; open var noWrap: Boolean; open var vAlign: String; open var bgColor: String companion object { val ELEMENT_NODE: Short val ATTRIBUTE_NODE: Short val TEXT_NODE: Short val CDATA_SECTION_NODE: Short val ENTITY_REFERENCE_NODE: Short val ENTITY_NODE: Short val PROCESSING_INSTRUCTION_NODE: Short val COMMENT_NODE: Short val DOCUMENT_NODE: Short val DOCUMENT_TYPE_NODE: Short val DOCUMENT_FRAGMENT_NODE: Short val DOCUMENT_POSITION_DISCONNECTED: Short val DOCUMENT_POSITION_PRECEDING: Short val DOCUMENT_POSITION_FOLLOWING: Short val DOCUMENT_POSITION_CONTAINS: Short val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Exposes the JavaScript [HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin

A public external abstract class HTMLFormElement: HTMLElement { open var acceptCharset: String; open var action: String; open var autocomplete: String; open var enctype: String; open var encoding: String; open var method: String; open var name: String; open var noValidate: Boolean; open var target: String; open val elements: HTMLFormControlsCollection; open val length: Int; fun submit(): Boolean; fun reportValidity(): Boolean companion object { val ELEMENT_NODE: Short val attributeNode: Short val TEXT_NODE: Short val CDATA_SECTION_NODE: Short val ENTITY_REFERENCE_NODE: Short val ENTITY_NODE: Short val PROCESSING_INSTRUCTION_NODE: Short val COMMENT_NODE: Short val DOCUMENT_NODE: Short val DOCUMENT_TYPE_NODE: Short val DOCUMENT_FRAGMENT_NODE: Short val DOCUMENT_POSITION_DISCONNECTED: Short val DOCUMENT_POSITION_PRECEDING: Short val DOCUMENT_POSITION_FOLLOWING: Short val DOCUMENT_POSITION_CONTAINS: Short val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Inline only operator fun HTMLFormElement.get(index: Int): Element? = asDynamic()[index]

Inline only operator fun HTMLFormElement.get(name: String): UnionElementOrRadioNodeList? = asDynamic()[name]

Exposes the JavaScript [HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin


DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short


DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin
 */
public external abstract class HTMLInputElement : HTMLElement {
    open var accept: String
    open var alt: String
    open var autocomplete: String
    open var autofocus: Boolean
    open var defaultChecked: Boolean
    open var dirName: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var height: Int
    open var indeterminate: Boolean
    open var inputMode: String
    open var list: HTMLElement?
    open var max: String
    open var maxLength: Int
    open var min: String
    open var minLength: Int
    open var multiple: boolean
    open var name: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var size: Int
    open var src: String
    open var step: String
    open var type: String
    open var defaultValue: String
    open var valueAsDate: dynamic
    open var valueAsNumber: Double
    open var width: Int
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    fun select()
    fun setRangeText(replacement: String)
    fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
    fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin
 */
public external abstract class HTMLButtonElement : HTMLElement {
    open var autofocus: Boolean
    open var disabled: Boolean
    open var form: HTMLFormElement?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var name: String
    open var type: String
    open var value: String
    open var menu: HTMLMenuElement?
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin
 */
public
external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {
    open var autocomplete: String?
    open var autofocus: Boolean
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var multiple: Boolean
    open var name: String
    open var required: Boolean
    open var size: Int
    open val type: String
    open val options: HTMLOptionsCollection
    override var length: Int
    open val selectedOptions: HTMLCollection
    open var selectedIndex: Int
    open var value: String
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun namedItem(name: String): HTMLOptionElement?
    fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
    fun remove(index: Int)
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    override fun item(index: Int): Element?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.set(index: Int, option: HTMLOptionElement?) = asDynamic()[index] = option

/**
 * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin
 */
public external abstract class HTMLDataListElement : HTMLElement {
    open val options: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin
 */
public external abstract class HTMLOptGroupElement : HTMLElement,

**Open Source Used In DNAC Platform Arica**

- [UnionHTMLOptGroupElementOrHTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/UnionHTMLOptGroupElementOrHTMLOptionElement)
- [HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement)
- [HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement)
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin
 */
public external abstract class HTMLOutputElement : HTMLElement {
    open val htmlFor: DOMTokenList
    open val form: HTMLFormElement?
    open var name: String
    open var type: String
    open var defaultValue: String
    open var value: String
    open val willValidate: Boolean
    open val validity:ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        val NODE_ITERATION_USE擅自*: Short
    }
}

/**
 * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin
 */
public external abstract class HTMLProgressElement : HTMLElement {
    open var value: Double
    open var max: Double
    open val position: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        val NODE_ITERATION_USE擅自*: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin
 */
public external abstract class HTMLMeterElement : HTMLElement {
    open var value: Double
    open var min: Double
    open var max: Double
    open var low: Double
    open var high: Double
    open var optimum: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        val NODE_ITERATION_USE擅自*: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin
 */
public external abstract class HTMLFieldSetElement : HTMLElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var name: String
    open val type: String
    open val elements: HTMLCollection
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        val NODE_ITERATION_USE擅自*: Short
    }
}
ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin
 */
public external abstract class HTMLLegendElement : HTMLElement {
    open val form: HTMLFormElement?
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin
 */
public external abstract class ValidityState {
    open val valueMissing: Boolean
    open val typeMismatch: Boolean
    open val patternMismatch: Boolean
    open val tooLong: Boolean
    open val tooShort: Boolean
    open val rangeUnderflow: Boolean
    open val rangeOverflow: Boolean
    open val stepMismatch: Boolean
    open val badInput: Boolean
    open val customError: Boolean
    open val valid: Boolean
}

/**
 * Exposes the JavaScript [HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to
Kotlin
 */
public external abstract class HTMLDetailsElement : HTMLElement {
    open var open: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLMenuElement : HTMLElement {
    open var type: String
    open var label: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLMenuItemElement : HTMLElement {
    open var type: String
    open var label: String
    open var icon: String
    open var disabled: Boolean
    open var checked: Boolean
    open var radiogroup: String


DocumentFragment

companion object {

val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLSlotElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSlotElement) to Kotlin */
public external abstract class HTMLSlotElement : HTMLElement {

open var name: String
fun assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>

companion object {

val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external interface AssignedNodesOptions {

var flatten: Boolean? /* = false */

get() = definedExternally

set(value) = definedExternally

fun assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>

}

@kotlin.internal.InlineOnly
public inline fun AssignedNodesOptions(flatten: Boolean? = false): AssignedNodesOptions {

val o = js("({})")

o["flatten"] = flatten

return o

}

/**
 * Exposes the JavaScript [HTMLCanvasElement](https://developer.mozilla.org/en/docs/Web/API/HTMLCanvasElement) to Kotlin */
public external abstract class HTMLCanvasElement : HTMLElement, CanvasImageSource, TexImageSource {

open var width: Int
open var height: Int

fun getContext(contextId: String, vararg arguments: Any?): RenderingContext?

fun toDataURL(type: String = definedExternally, quality: Any? = definedExternally): String

fun toBlob(_callback: (Blob?) -> Unit, type: String = definedExternally, quality: Any? = definedExternally)

companion object {

val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external interface CanvasRenderingContext2DSettings {

var alpha: Boolean? /* = true */

get() = definedExternally

set(value) = definedExternally

fun getContext2DSettings(alpha: Boolean? = true): CanvasRenderingContext2DSettings

}

@kotlin.internal.InlineOnly
public inline fun CanvasRenderingContext2DSettings(alpha: Boolean? = true): CanvasRenderingContext2DSettings {

val o = js("({})")

o["alpha"] = alpha

return o

}

/**
 * Exposes the JavaScript [CanvasRenderingContext2D](https://developer.mozilla.org/en/docs/Web/API/CanvasRenderingContext2D) to Kotlin */
public *unpublic external abstract class CanvasRenderingContext2D : CanvasState, CanvasTransform, CanvasCompositing, CanvasImageSmoothing, CanvasFillStrokeStyles, CanvasShadowStyles, CanvasFilters, CanvasRect, CanvasDrawPath, CanvasUserInterface, CanvasText, CanvasDrawImage, CanvasHitRegion,
CanvasImageData, CanvasPathDrawingStyles, CanvasTextDrawingStyles, CanvasPath, RenderingContext

open val canvas: HTMLCanvasElement

public external interface CanvasState {
    fun save()
    fun restore()
}

public external interface CanvasTransform {
    fun scale(x: Double, y: Double)
    fun rotate(angle: Double)
    fun translate(x: Double, y: Double)
    fun setTransform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)
    fun getTransform(): DOMMatrix
    fun setTransform(transform: dynamic)
    fun resetTransform()
}

public external interface CanvasCompositing {
    var globalAlpha: Double
    var globalCompositeOperation: String
}

public external interface CanvasImageSmoothing {
    var imageSmoothingEnabled: Boolean
    var imageSmoothingQuality: ImageSmoothingQuality
}

public external interface CanvasFillStrokeStyles {
    var strokeStyle: dynamic
    var fillStyle: dynamic
    fun createLinearGradient(x0: Double, y0: Double, x1: Double, y1: Double)
    fun createRadialGradient(x0: Double, y0: Double, x1: Double, y1: Double)
    fun createPattern(image: CanvasImageSource, repetition: String)
    fun createGradient(sx: Double, sy: Double, sw: Double, sh: Double)
}

public external interface CanvasShadowStyles {
    var shadowOffsetX: Double
    var shadowOffsetY: Double
    var shadowBlur: Double
    var shadowColor: String
}

public external interface CanvasFilters {
    var filter: String
}

public external interface CanvasRect {
    fun clearRect(x: Double, y: Double, w: Double, h: Double)
    fun fillRect(x: Double, y: Double, w: Double, h: Double)
    fun strokeRect(x: Double, y: Double, w: Double, h: Double)
}

public external interface CanvasDrawPath {
    fun beginPath()
    fun fill(fillRule: CanvasFillRule)
    fun fill(path: Path2D, fillRule: CanvasFillRule)
    fun stroke(path: Path2D)
    fun clip(fillRule: CanvasFillRule)
    fun createPath(fillRule: CanvasFillRule)
    fun resetClip()
    fun isPointInPath(x: Double, y: Double, fillRule: CanvasFillRule)
    fun isPointInPath(path: Path2D, x: Double, y: Double, fillRule: CanvasFillRule)
    fun isPointInStroke(x: Double, y: Double)
    fun isPointInStroke(path: Path2D, x: Double, y: Double)
}

public external interface CanvasUserInterface {
    fun drawFocusIfNeeded(element: Element)
    fun drawFocusIfNeeded(path: Path2D, element: Element)
}

public external interface CanvasText {
    fun fillText(text: String, x: Double, y: Double, maxWidth: Double)
    fun strokeText(text: String, x: Double, y: Double, maxWidth: Double)
    fun measureText(text: String)
}

public external interface CanvasDrawImage {
    fun drawImage(image: CanvasImageSource, dx: Double, dy: Double)
    fun drawImage(image: CanvasImageSource, sx: Double, sy: Double, sw: Double, sh: Double)
}

public external interface CanvasHitRegion {
    fun addHitRegion(options: HitRegionOptions)
    fun removeHitRegion(id: String)
}

public external interface CanvasImageData {
    fun createImageData(sw: Double, sh: Double)
    fun getImageData(sx: Double, sy: Double, sw: Double, sh: Double)
}

public external interface CanvasPathDrawingStyles {
    var lineWidth: Double
    var lineCap: CanvasLineCap
    var lineJoin: CanvasLineJoin
    var miterLimit: Double
    fun setLineDash(segments: Array<Double>)
    fun getLineDash(): Array<Double>
}

public external interface CanvasTextDrawingStyles {
    var font: String
    var textAlign: CanvasTextAlign
    var textBaseline: CanvasTextBaseline
    var direction: String
}

Open Source Used In DNAC Platform Arica 12318
/**
 * Exposes the JavaScript [Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin
 */

public external abstract class Window : EventTarget, GlobalEventHandlers, WindowEventHandlers, WindowOrWorkerGlobalScope, WindowSessionStorage, WindowLocalStorage, GlobalPerformance, UnionMessagePortOrWindowProxy

/**
 * Exposes the JavaScript [Window](https://developer.mozilla.org/en/docs/Web/API/Window) to Kotlin
 */

public interface WindowOrWorkerGlobalScope {
    open val window: Window
    open val self: Window
    open val document: Document
    open var name: String
    open val location: Location
    open val history: History
    open val forward: Array<Window>
    open val frames: Window
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
    open val length: Int
    open val top: Window
    open var opener: Any?
    open val parent: Window
    open val frameElement: Element
    open val navigator: Navigator
    open val applicationCache: ApplicationCache
    open val external: External
    open val screen: Screen
    open val locationbar: BarProp
    open val menubar: BarProp
    open val personalbar: BarProp
    open val scrollbars: BarProp
    open val statusbar: BarProp
    open val toolbar: BarProp
    open var status: String
    open val closed: Boolean
    open val frames: Array<Window>
Event | open val oldURL: String
open val newURL: String
companion object { val NONE:
val CAPTURING_PHASE: Short
val AT_TARGET: Short
val BUBBLING_PHASE:
|\n\npublic external interface HashChangeEventInit : EventInit { val var oldURL: String
* = \"\" */\nget() = definedExternally
set(value) = definedExternally
var newURL: String
* = \"\" */\nget() = definedExternally
set(value) = definedExternally
}@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly

*/
public external open class PromiseRejectionEvent(type: String, eventInitDict: PromiseRejectionEventInit) : Event {
    open val promise: Promise<Any?>
    open val reason: Any?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface PromiseRejectionEventInit : EventInit {
    var promise: Promise<Any?>?
        get() = definedExternally
        set(value) = definedExternally
    var reason: Any?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["promise"] = promise
    o["reason"] = reason
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}**

/* Exposes the JavaScript [GlobalEventHandlers](https://developer.mozilla.org/en/docs/Web/API/GlobalEventHandlers) to Kotlin*/

/*
* public external interface GlobalEventHandlers {
var onabort: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onblur: ((FocusEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oncancel: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oncanplay: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oncanplaythrough: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onclick: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onclose: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oncontextmenu: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oncuechange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondblclick: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondrag: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondragend: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondragenter: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondragexit: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondragover: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondragstart: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondrop: ((DragEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var ondurationchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onemptied: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onfocus: ((FocusEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oninput: ((InputEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var oninvalid: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onkeydown: ((KeyboardEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onkeypress: ((KeyboardEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onkeyup: ((KeyboardEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmousedown: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmousemove: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmouseout: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmouseover: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmouseup: ((MouseEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onmoveend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onresizestart: ((ProgressEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onresizeend: ((ProgressEvent) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectstart: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionstart: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var selectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionchange: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)?
  get() =
definedExternally
  set(value) = definedExternally
var onselectionend: ((Event) -> dynamic)
definedExternally
set(value) = definedExternally
var onmouseenter: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmouseleave: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmousemove: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmouseout: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmouseover: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmouseup: ((MouseEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onwheel: ((WheelEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpause: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onplay: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onplaying: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onprogress: ((ProgressEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onratechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onreset: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var on resize: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsubmit: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsuspend: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onvolumechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onwaiting: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var ong pointercapture: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointerdown: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointerup: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointerleave: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeprint: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onhashchange: ((HashChangeEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onlanguagechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onmessage: ((MessageEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onoffline: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var ononline: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onorientationchange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeprint: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onhashchange: ((HashChangeEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onlanguagechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onorientationchange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onprogress: ((ProgressEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onratechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onreset: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onresize: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsubmit: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsuspend: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onvolumechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onwaiting: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var ong pointercapture: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointerleave: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeprint: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onhashchange: ((HashChangeEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onlanguagechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onorientationchange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onprogress: ((ProgressEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onratechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onreset: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onresize: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsubmit: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onsuspend: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onvolumechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onwaiting: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var ong pointercapture: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onpointerleave: ((PointerEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeprint: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onbeforeunload: ((BeforeUnloadEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onhashchange: ((HashChangeEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onlanguagechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onorientationchange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onprogress: ((ProgressEvent) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onratechange: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onreset: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onresize: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
get() = definedExternally
set(value) = definedExternally
var onselect: ((Event) -> dynamic)
definedExternally
var ononline: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononpagehide: ((PageTransitionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononpageshow: ((PageTransitionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononpopstate: ((PopStateEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononrejectionhandled: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononstorage: ((StorageEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var ononunload: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

}

public external interface DocumentAndElementEventHandlers {
    var oncopy: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncut: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpaste: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript [WindowOrWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WindowOrWorkerGlobalScope) to Kotlin
 */
public external interface WindowOrWorkerGlobalScope {
    val origin: String
    val caches: CacheStorage
    fun btoa(data: String): String
    fun atob(data: String): String
    fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearTimeout(handle: Int = definedExternally)
    fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearInterval(handle: Int = definedExternally)
    fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>
    fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>
}

/**
 * Exposes the JavaScript [Navigator](https://developer.mozilla.org/en/docs/Web/API/Navigator) to Kotlin
 */
public external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {
    open val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val maxTouchPoints: Int
    open val serviceWorker: ServiceWorkerContainer
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

/**
 * Exposes the JavaScript [NavigatorID](https://developer.mozilla.org/en/docs/Web/API/NavigatorID) to Kotlin
 */
public external interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val platform: String
    val product: String
    val productSub: String
    val userAgent: String
    val vendor: String
    val vendorSub: String
    val oscpu: String
    fun taintEnabled(): Boolean
}

/**
 * Exposes the JavaScript [NavigatorLanguage](https://developer.mozilla.org/en/docs/Web/API/NavigatorLanguage) to Kotlin
 */
public external interface NavigatorLanguage {
    val language: String
    val languages: Array<out String>
}

public external interface NavigatorContentUtils {
    fun registerProtocolHandler(scheme: String, url: String, title: String)
    fun registerContentHandler(mimeType: String, url: String, title: String)
    fun isProtocolHandlerRegistered(scheme: String, url: String): String
    fun isContentHandlerRegistered(mimeType: String, url: String): String
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

/**
 * Exposes the JavaScript [NavigatorPlugins](https://developer.mozilla.org/en/docs/Web/API/NavigatorPlugins) to Kotlin
 */
public external abstract class NavigatorPlugins : PluginArray {
    fun refresh(reload: Boolean = definedExternally)
    fun unregisterContentHandler(mimeType: String, url: String)
    val plugins: PluginArray
    val mimeTypes: MimeTypeArray
    fun javaEnabled(): Boolean
}

@kotlin.internal.InlineOnly
inline operator fun PluginArray.get(index: Int): Plugin? =
asDynamic()[index]@kotlin.internal.InlineOnly\npublic inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]@kotlin.internal.InlineOnly\n
[MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin\n
abstract class MimeTypeArray : ItemArrayLike<MimeType> {
  override fun item(index: Int): MimeType? = asDynamic()[index]

  fun namedItem(name: String): MimeType?
}

abstract class Plugin : ItemArrayLike<MimeType> {
  open val name: String
  open val description: String
  open val filename: String
  override fun item(index: Int): MimeType? = asDynamic()[index]

  fun namedItem(name: String): MimeType?
}

abstract class MimeType {
  open val type: String
  open val description: String
  open val suffixes: String
  open val enabledPlugin: Plugin
}

abstract class ImageBitmap : CanvasImageSource, TexImageSource {
  open val width: Int
  open val height: Int
  fun close()
}

interface ImageBitmapOptions {
  var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */
  var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */
  var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */
  var resizeWidth: Int?
  var resizeHeight: Int?
  var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */
}

  val o = js("({})")
  o["imageOrientation"] = imageOrientation?
  o["premultiplyAlpha"] = premultiplyAlpha?
  o["colorSpaceConversion"] = colorSpaceConversion?
  o["resizeWidth"] = resizeWidth?
  o["resizeHeight"] = resizeHeight?
  o["resizeQuality"] = resizeQuality?
  return o?
}

abstract class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {
  var open val data: Any?
  var open val lastEventId: String?
  var open val source: UnionMessagePortOrWindowProxy?
  var open val ports: Array<out MessagePort>?
  fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>?): MessageEventInit = definedExternally:

  Event {
    var open val data: Any?
    var open val lastEventId: String?
    var open val source: UnionMessagePortOrWindowProxy?
    var open val ports: Array<MessagePort>?
  }

  override fun item(index: Int): MimeType? = asDynamic()[index]

  fun namedItem(name: String): MimeType?
  fun namedItem(name: String): MimeType?
  override fun item(index: Int): MimeType? = asDynamic()[index]

  fun namedItem(name: String): MimeType?
  override fun item(index: Int): MimeType? = asDynamic()[index]

  fun namedItem(name: String): MimeType?
  override fun item(index: Int): MimeType? = asDynamic()[index]
definedExternally

Open Source Used In DNAC Platform Arica  12327

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin */
public external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
    open val url: String
    open val withCredentials: Boolean
    open val readyState: Short
    var onopen: ((Event) -> dynamic)?
    var onmessage: ((MessageEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    fun close()

    companion object {
        val CONNECTING: Short
        val OPEN: Short
        val CLOSED:
    }

    public external interface EventSourceInit : EventInit {
        var withCredentials: Boolean? /* = false */

        get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
        val o = js("({})")
        o["withCredentials"] = withCredentials
        return o
    }

    /**
     * Exposes the JavaScript [WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin */
    public external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
        open val url: String
        open val readyState: Short
        open val bufferedAmount: Number
        var onopen: ((Event) -> dynamic)?
        var onerror: ((Event) -> dynamic)?
        var onclose: ((Event) -> dynamic)?
        open val extensions: String
        open val protocol: String
        var onmessage: ((MessageEvent) -> dynamic)?
        var binaryType: BinaryType
        fun close(code: Short = definedExternally, reason: String = definedExternally)
        fun send(data: String)
        fun send(data: Blob)
        fun send(data: ArrayBuffer)
        fun send(data: ArrayBufferView)

        companion object {
            val CONNECTING:
            val OPEN: Short
            val CLOSING: Short
            val CLOSED: Short
        }
    }

    /**
     * Exposes the JavaScript [CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin
     */
    public external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {
        open val wasClean: Boolean
        open val code: Short
        open val reason: String

        companion object {
            val NONE:
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE:
        }

    public external interface CloseEventInit : EventInit {
        var wasClean: Boolean? /* = false */

        get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["wasClean"] = wasClean
        o["code"] = code
        o["reason"] = reason
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /**
     * Exposes the JavaScript [MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin */
    public external open class MessageChannel {
        open val port1: MessagePort
        open val port2: MessagePort
    }

    /**
     * Exposes the JavaScript [MessagePort](https://developer.mozilla.org/en/docs/Web/API/MessagePort) to Kotlin */
    public external abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy, UnionClientOrMessagePortOrServiceWorker, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
        open var onmessage: ((MessageEvent) -> dynamic)?
        fun postMessage(message: Any?, transfer: ArrayBuffer, arrayBufferView: ArrayBufferView)
        fun set(name: String)
    }

    /**
     * Exposes the JavaScript [BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin */
    public external open class BroadcastChannel(name: String) : EventTarget, UnionMessagePortOrWindowProxy, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
        open var open value: WorkerGlobalScope
        open var open value: WorkerGlobalScope
    }

    /**
     * Exposes the JavaScript [WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin */
    public external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope, GlobalPerformance {
        open var open value: WorkerGlobalScope
        open var open value: WorkerGlobalScope
    }
```
navigator: WorkerNavigator
    open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    open var onlanguagechange: ((Event) -> dynamic)?
    open var onoffline: ((Event) -> dynamic)?
    open var ononlinestatechange: ((Event) -> dynamic)?
    open var onrejectionhandled: ((PromiseRejectionEvent) -> dynamic)?
    fun importScripts(vararg urls: String)
}

/**
 * Exposes the
 * Exposes the JavaScript
 */
public external abstract class DedicatedWorkerGlobalScope : WorkerGlobalScope {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun close()
}

/**
 * Exposes the JavaScript
 */
public external abstract class SharedWorkerGlobalScope : WorkerGlobalScope {
    open val name: String
    open val applicationCache: ApplicationCache
    open var onconnect: ((Event) -> dynamic)?
    fun close()
}

/**
 * Exposes the JavaScript
 */
public external abstract class AbstractWorker {
    var onerror: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
 */
public external open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    var onmessage: ((MessageEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    fun terminate()
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

public external interface WorkerOptions {
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
    var credentials: RequestCredentials? /* = RequestCredentials.OMIT */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun WorkerOptions(type: WorkerType? = WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {
    val o = js("({})")
    o["type"] = type
    o["credentials"] = credentials
    return o
}

/**
 * Exposes the JavaScript
 */
public external open class SharedWorker(scriptURL: String, name: String = definedExternally, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    open val port: MessagePort
    override var onerror: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript
 */
public external interface NavigatorConcurrentHardware {
    val hardwareConcurrency: Number
}

/**
 * Exposes the JavaScript
 */
public external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorConcurrentHardware {
    open val serviceWorker: ServiceWorkerContainer
}

/**
 * Exposes the JavaScript
 */
public external abstract class WorkerLocation {
    open val href: String
    open val origin: String
    open val protocol: String
    open val hostname: String
    open val port: String
    open val pathname: String
    open val search: String
    open val hash: String
    open val fullpath: String
    open val fullpathWithHash: String
    fun getItem(key: String): String?
    fun setItem(key: String, value: String): String?
    fun removeItem(key: String)
    fun clear()
}

@kotlin.internal.InlineOnly
public inline operator fun Storage.get(key: String): String? = asDynamic()[key]

@kotlin.internal.InlineOnly
public inline operator fun Storage.set(key: String, value: String) { asDynamic()[key] = value }

/**
 * Exposes the JavaScript
 */
public external interface WindowSessionStorage {
    val sessionStorage: Storage
}

/**
 * Exposes the JavaScript
 */
public external abstract class WindowSessionStorage {
    val sessionStorage: Storage
}

Open Source Used In DNAC Platform Arica 12328
JavaScript [WindowLocalStorage](https://developer.mozilla.org/en/docs/Web/API/WindowLocalStorage) to Kotlin

```kotlin
@public external interface WindowLocalStorage {
    val localStorage: Storage
}

/**
 * Exposes the JavaScript [StorageEvent](https://developer.mozilla.org/en/docs/Web/API/StorageEvent) to Kotlin
 */
public external open class StorageEvent(type: String, eventInitDict: StorageEventInit = definedExternally) : Event {
    open var key: String?
    open var oldValue: String?
    open var newValue: String?
    open var url: String?
    open var storageArea: Storage?
    open val NONE: Short
    open val AT_TARGET: Short
    open val BUBBLING_PHASE: Short
    open var public external interface StorageEventInit : EventInit {
        var key: String? /* = null */
        var oldValue: String? /* = null */
        var newValue: String? /* = null */
        var url: String? /* = "" */
        var storageArea: Storage? /* = null */
    }
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["key"] = key
    o["oldValue"] = oldValue
    o["newValue"] = newValue
    o["url"] = url
    o["storageArea"] = storageArea
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    return o
}

@public external abstract class HTMLAppletElement : HTMLElement {
    open var align: String
    open var alt: String
    open var archive: String
    open var code: String
    open var codeBase: String
    open var height: String
    open var hspace: Int
    open var name: String
    open var _object: String
    open var vspace: Int
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin
 */
public external abstract class HTMLMarqueeElement : HTMLElement {
    open var behavior: String
    open var bgColor: String
    open var direction: String
    open var height: String
    open var hspace: Int
    open var loop: Int
    open var scrollAmount: Int
    open var scrollDelay: Int
    open var trueSpeed: Boolean
    open var vspace: Int
    open var width: String
    open fun start()
    open fun stop()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin
 */
public external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {
    open var cols: String
    open var rows: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the
 * JavaScript
 * [CustomEvent](https://developer.mozilla.org/en/docs/Web/API/CustomEvent) to Kotlin
 */
public external open class CustomEvent(type: String, eventInitDict: CustomEventInit = definedExternally) : Event {
    open val detail: Any?
    fun initCustomEvent(type: String, bubbles: Boolean, cancelable: Boolean, detail: Any?)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CustomEventInit : EventInit {
    var detail: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

@kotlin.internal.InlineOnly
public inline fun EventListenerOptions(capture: Boolean? = false): EventListenerOptions {
    val o = js("({})")
    o["capture"] = capture
    return o
}

public external interface AddEventListenerOptions : EventListenerOptions {
    var passive: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var once: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["passive"] = passive
    o["once"] = once
    return o
}

@kotlin.internal.InlineOnly
public inline fun NodeArrayLike<*> get(index: Int): Node?

@kotlin.internal.InlineOnly
public inline fun NodeArrayLike<*> get(name: String): Node?

@kotlin.internal.InlineOnly
public inline fun HTMLCollection.get(index: Int): Node?

@kotlin.internal.InlineOnly
public inline fun HTMLCollection.get(name: String): Node?

Open Source Used In DNAC Platform Arica 12331
String): Element? = asDynamic()[name]

/**
 * Exposes the JavaScript [MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin
 */
public open class MutationObserver(callback: (Array<MutationRecord>, MutationObserver) -> Unit) {

    fun observe(target: Node, options: MutationObserverInit = definedExternally)
    fun disconnect()
    fun takeRecords(): Array<MutationRecord>
}

/**
 * Exposes the JavaScript [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {

    var childList: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtrees: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly

    val o = js("({})")

    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    o["subtrees"] = subtrees
    o["attributeOldValue"] = attributeOldValue
    o["characterDataOldValue"] = characterDataOldValue
    o["attributeFilter"] = attributeFilter

    return o
}

/**
 * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin
 */
public external abstract class Node : EventTarget {

    open var nodeType: Short
    open var nodeName: String
    open var baseURI: String
    open var isConnected: Boolean
    open var ownerDocument: Document?
    open var parentNode: Node?
    open var parentElement: Element?
    open var childNodes: NodeList
    open var firstChild: Node?
    open var lastChild: Node?
    open var previousSibling: Node?
    open var nextSibling: Node?

    open var nodeValue: String?
    open var textContent: String?

    fun getRootNode(options: GetRootNodeOptions = definedExternally): Node
    fun hasChildNodes(): Boolean
    fun normalize()
    fun cloneNode(deep: Boolean = definedExternally): Node
    fun replaceChild(node: Node, child: Node): Node
    fun removeChild(child: Node): Node

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface GetRootNodeOptions {

    var composed: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean?
Exposed the JavaScript `Document` class to Kotlin.

```kotlin
fun GetRootNodeOptions(\n    val o = js("({})")\n) = composed
```

Exposed the `Document` class:

```kotlin
    open val open origin: String
    open val open documentElement: Element?
    open val open location: Location?
    open val open domain: String
    open val open referrer: String
    open val open cookie: String
    open val open lastModified: String
    open val open readyState: DocumentReadyState
    open val open title: String
    open val open dir: String
    open val open body: HTMLElement?
    open val open head: HTMLHeadElement?
    open val open images: HTMLCollection
    open val open embeds: HTMLCollection
    open val open plugins: HTMLCollection
    open val open links: HTMLCollection
    open val open forms: HTMLCollection
    open val open scripts: HTMLCollection
    open val open currentScript: HTMLOrSVGScriptElement?
    open val open defaultView: Window?
    override val open abort: ((Event) -> dynamic)?
    override val open blur: ((FocusEvent) -> dynamic)?
    override val open canplay: ((Event) -> dynamic)?
    override val open canplaythrough: ((Event) -> dynamic)?
    override val open change: ((Event) -> dynamic)?
    override val open click: ((MouseEvent) -> dynamic)?
    override val open close: ((Event) -> dynamic)?
    override val opencontextmenu: ((MouseEvent) -> dynamic)?
    override val oncuechange: ((Event) -> dynamic)?
    override val ondblclick: ((MouseEvent) -> dynamic)?
    override val ondrag: ((DragEvent) -> dynamic)?
    override val ondragend: ((DragEvent) -> dynamic)?
    override val ondragenter: ((DragEvent) -> dynamic)?
    override val ondragleave: ((DragEvent) -> dynamic)?
    override val ondragover: ((DragEvent) -> dynamic)?
    override val ondragstart: ((DragEvent) -> dynamic)?
    override val ondurationchange: ((Event) -> dynamic)?
    override val onemptied: ((Event) -> dynamic)?
    override val onended: ((Event) -> dynamic)?
    override val onerror: ((Any?) -> dynamic)?
    override val onfocus: ((FocusEvent) -> dynamic)?
    override val oninput: ((InputEvent) -> dynamic)?
    override val onkeydown: ((KeyboardEvent) -> dynamic)?
    override val onkeypress: ((KeyboardEvent) -> dynamic)?
    override val onkeyup: ((KeyboardEvent) -> dynamic)?
    override val onload: ((Event) -> dynamic)?
    override val onloaddata: ((Event) -> dynamic)?
    override val onreadystatechange: ((Event) -> dynamic)?
    override val onreset: ((Event) -> dynamic)?
    override val onresize: ((Event) -> dynamic)?
    override val onscroll: ((Event) -> dynamic)?
    override val onselect: ((Event) -> dynamic)?
    override val onshow: ((Event) -> dynamic)?
    override val onstalled: ((Event) -> dynamic)?
    override val onsubmit: ((Event) -> dynamic)?
    override val ontimeupdate: ((Event) -> dynamic)?
    override val ontoggle: ((Event) -> dynamic)?
    override val var
```
onvolumechange: ((Event) -> dynamic)?
override var onwaiting: ((Event) -> dynamic)?
override var ongpointercapture: ((PointerEvent) -> dynamic)?
override var onpointerdown: ((PointerEvent) -> dynamic)?
override var onpointermove: ((PointerEvent) -> dynamic)?
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerout: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy: ((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override var fullScreenElement: Element?
override var children: HTMLCollection
override var firstElementChild: Element?
override var lastElementChild: Element?
override var childElementCount: Int
fun getElementsByTagName(qualifiedName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun createElementNS(namespace: String?, qualifiedName: String, options: ElementCreationOptions = definedExternally): Element
fun createDocumentFragment(): DocumentFragment
fun createTextNode(data: String): Text
fun createCDATASection(data: String): CDATASection
fun createComment(data: String): Comment
fun createProcessingInstruction(target: String, data: String): ProcessingInstruction
fun importNode(node: Node, deep: Boolean = definedExternally): Node?
fun adoptNode(node: Node): Node?
fun createAttribute(localName: String?): Attr
fun createAttributeNS(namespace: String?, qualifiedName: String): Attr
fun createEvent(`interface`: String): Event
fun createRange(): Range
fun createTreeWalker(root: Node, whatToShow: Int = definedExternally, filter: NodeFilter? = definedExternally): TreeWalker
fun getElementById(elementId: String): Element?
fun prepend(vararg nodes: dynamic)
fun append(vararg nodes: dynamic)
fun querySelector(selectors: String): Element?
fun querySelectorAll(selectors: String): NodeList
fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMRect
fun companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
}
Open Source Used In DNAC Platform Arica  12336

Short

val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [ShadowRoot](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) to Kotlin
 */
public external open

class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {
    open val mode: ShadowRootMode

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin
 */
public external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils, UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent, UnionElementOrProcessingInstruction {
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val tagName: String
    open var id: String
    open var className: String
    open val classList: DOMTokenList
    open var slot: String
    open val attributes: NamedNodeMap
    open val shadowRoot: ShadowRoot?
    open var scrollTop: Double
    open var scrollLeft: Double
    open val scrollWidth: Int
    open val scrollHeight: Int
    open val clientTop: Int
    open val clientLeft: Int
    open val clientWidth: Int
    open val clientHeight: Int
    open var innerHTML: String
    open var outerHTML: String

    fun hasAttributes(): Boolean
    fun getAttributeNames(): Array<String>
    fun getAttribute(qualifiedName: String): String?
    fun getAttributeNS(namespace: String?, localName: String): String?
    fun setAttribute(qualifiedName: String, value: String)
    fun setAttributeNS(namespace: String?, qualifiedName: String, value: String)
    fun removeAttribute(qualifiedName: String)
    fun removeAttributeNS(namespace: String?, localName: String)
    fun hasAttribute(qualifiedName: String): Boolean
    fun hasAttributeNS(namespace: String?, localName: String): Boolean
    fun getAttributeNode(qualifiedName: String): Attr?
    fun getAttributeNodeNS(namespace: String?, localName: String): Attr?
    fun setAttributeNode(attr: Attr): Attr?
    fun setAttributeNodeNS(attr: Attr): Attr?
    fun attachShadow(init: ShadowRootInit): ShadowRoot
    fun closest(selectors: String): Element?
    fun matches(selectors: String): Boolean
    fun webkitMatchesSelector(selectors: String): Boolean

    fun getElementsByTagName(qualifiedName: String): HTMLCollection
    fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
    fun getElementsByClassName(classNames: String): HTMLCollection
    fun getElementsByTagNameNS(): HTMLCollection
    fun insertAdjacentElement(where: String, element: Element): Element?
    fun insertAdjacentText(where: String, data: String): Element?
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun scrollIntoView()
    fun scrollIntoView(arg: dynamic): Element?
    fun scrollBy(options: ScrollToOptions = definedExternally): Element?
    fun scrollBy(options: ScrollToOptions = definedExternally): Element?
    fun scrollBy(x: Double, y: Double): Element?
    fun scrollBy(options: ScrollToOptions = definedExternally): Element?
    fun scrollBy(x: Double, y: Double): Element?
    fun insertAdjacentHTML(position: String, text: String): Element?
    fun setPointerCapture(pointerId: Int): Boolean
    fun releasePointerCapture(pointerId: Int): Boolean
    fun requestFullscreen():
String = definedExternally) : CharacterData, Slotable, GeometryUtilities {
    open val wholeText: String
    override val assignedSlot: HTMLSlotElement?
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    fun splitText(offset: Int): Text
    override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
    override fun convertQuadFromNode(quad: DOMQuad, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin
 */
public external open class CDATASection : Text {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin
 */
public external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {
    open val target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin
 */
public external open class Comment(data: String = definedExternally) : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {
    open val previousElementSibling: Element?
    override val nextElementSibling: Element?
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}
}

/**
 * Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
 */
public external open class Range {
    open val startContainer: Node
    open val startOffset: Int
    open val endContainer: Node
    open val endOffset: Int
    open val collapsed: Boolean
    open val commonAncestorContainer: Node
    fun setStart(node: Node, offset: Int)
    fun setEnd(node: Node, offset: Int)
    fun setStartBefore(node: Node)
    fun setStartAfter(node: Node)
    fun setEndBefore(node: Node)
    fun setEndAfter(node: Node)
    fun collapse(toStart: Boolean = definedExternally)
    fun selectNode(node: Node)
    fun selectNodeContents(node: Node)
    fun compareBoundaryPoints(how: Short, sourceRange: Range): Short
    fun deleteContents()
    fun extractContents(): DocumentFragment
    fun cloneContents(): DocumentFragment
    fun insertNode(node: Node)
    fun surroundContents(newParent: Node)
    fun cloneRange(): Range
    fun detach()
    fun isPointInRange(node: Node, offset: Int): Boolean
    fun comparePoint(node: Node, offset: Int): Short
    fun intersectsNode(node: Node)
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun createContextualFragment(fragment: String): DocumentFragment

    companion object {
        val START_TO_START: Short
        val START_TO_END: Short
        val END_TO_END: Short
        val END_TO_START: Short
    }
}

/**
 * Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
    open val root: Node
    open val referenceNode: Node
    open val pointerBeforeReferenceNode: Boolean
    open val whatToShow: Int
    open val filter: NodeFilter?
    fun nextNode(): Node?
    fun previousNode(): Node?
    fun detach()
}

/**
 * Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
    open val root: Node
    open val whatToShow: Int
    open val filter: NodeFilter?
    open var currentNode: Node
    fun parentNode(): Node?
    fun firstChild(): Node?
    fun lastChild(): Node?
    fun previousSibling(): Node?
    fun nextSibling(): Node?
    fun previousNode(): Node?
    fun nextNode(): Node?
}

/**
 * Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
    fun acceptNode(node: Node): Short

    companion object {
        val FILTER_ACCEPT: Short
        val FILTER_REJECT: Short
        val FILTER_SKIP: Short
        val SHOW_ALL: Int
        val SHOW_ELEMENT: Int
        val SHOW_ATTRIBUTE: Int
        val SHOW_TEXT: Int
        val SHOW_CDATA_SECTION: Int
        val SHOW_ENTITY_REFERENCE: Int
        val SHOW_PROCESSING_INSTRUCTION: Int
    }
}

/**
 * Exposes the JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
 */
public external abstract class DOMTokenList : ItemArrayLike<String> {
    open var value: String
    fun contains(token: String): Boolean
    fun add(vararg tokens: String)
    fun remove(vararg tokens: String)
    fun toggle(token: String, force: Boolean = definedExternally): Boolean
    fun replace(token: String, newToken: String)
    fun supports(token: String): Boolean
    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript [DOMPointReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMPointReadOnly) to Kotlin
 */
public external open class DOMPointReadOnly(x: Double, y: Double, z: Double, w: Double) {
    open val x: Double

open val y: Double
open val z: Double
open val w: Double

fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint

/**
 * Exposes the JavaScript [DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin
 */
public external open class DOMPoint : DOMPointReadOnly {
    constructor(point: DOMPointInit)
    constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)

    override var x: Double
    override var y: Double
    override var z: Double
    override var w: Double
}

/**
 * Exposes the JavaScript [DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin
 */
public external interface DOMPointInit {
    var x: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var y: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var z: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var w: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["z"] = z
    o["w"] = w
    return o
}

/**
 * Exposes the JavaScript [DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin
 */
public external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {
    override var x: Double
    override var y: Double
    override var width: Double
    override var height: Double
}

/**
 * Exposes the JavaScript [DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin
 */
public external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {
    open val x: Double
    open val y: Double
    open val width: Double
    open val height: Double
    open val top: Double
    open val right: Double
    open val bottom: Double
    open val left: Double
}

public external interface DOMRectList : ItemArrayLike<DOMRect> {
    override fun item(index: Int): DOMRect?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]

/**
 * Exposes the JavaScript [DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin
 */
public external open class DOMQuad {
    constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)
    constructor(rect: DOMRectInit)

    open val p1: DOMPoint
    open val p2: DOMPoint
    open val p3: DOMPoint
    open val p4: DOMPoint

    open val bounds: DOMRectReadOnly
}

/**
 * Exposes the JavaScript [DOMMatrixReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMMatrixReadOnly) to Kotlin
 */
public external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {
    open val a: Double
    open val b: Double
    open val c: Double
    open val d: Double
    open val e: Double
    open val f: Double

    open val m11: Double
    open val m12: Double
    open val m13: Double
    open val m14: Double
    open val m21: Double
    open val m22: Double
    open val m23: Double
    open val m24: Double
    open val m31: Double
    open val m32: Double
    open val m33: Double
    open val m34: Double
    open val m41: Double
    open val m42: Double
    open val m43: Double
    open val m44: Double
    open val is2D: Boolean
    open val isIdentity: Boolean
    fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
Double = definedExternally): DOMMatrix
    fun scaleNonUniform(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun rotate(angle: Double, originX: Double, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVector(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngle(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
    fun skewX(sx: Double): DOMMatrix
    fun skewY(sy: Double): DOMMatrix
    fun multiply(other: DOMMatrix): DOMMatrix
    fun flipX(): DOMMatrix
    fun flipY(): DOMMatrix
    fun inverse(): DOMMatrix
    fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint
    fun toFloat32Array(): Float32Array
    fun toFloat64Array(): Float64Array
}

/**
 * Exposes the JavaScript [DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin
 */
public external open class DOMMatrix() : DOMMatrixReadOnly {
    constructor(transformList: String)
    constructor(other: DOMMatrixReadOnly)
    constructor(array32: Float32Array)
    constructor(array64: Float64Array)
    constructor(numberSequence: Array<Double>)
    override var a: Double
    override var b: Double
    override var c: Double
    override var d: Double
    override var e: Double
    override var f: Double
    override var m11: Double
    override var m12: Double
    override var m13: Double
    override var m14: Double
    override var m21: Double
    override var m22: Double
    override var m23: Double
    override var m24: Double
    override var m31: Double
    override var m32: Double
    override var m33: Double
    override var m34: Double
    override var m41: Double
    override var m42: Double
    override var m43: Double
    override var m44: Double
    fun multiplySelf(other: DOMMatrix): DOMMatrix
    fun preMultiplySelf(other: DOMMatrix): DOMMatrix
    fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3dSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVectorSelf(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngleSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
    fun skewXSelf(sx: Double): DOMMatrix
    fun skewYSelf(sy: Double): DOMMatrix
    fun invertSelf(): DOMMatrix
    fun setMatrixValue(transformList: String): DOMMatrix
}

public external interface ScrollOptions {
    var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {
    val o = js("({})")
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [ScrollToOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollToOptions) to Kotlin
 */
public external interface ScrollToOptions : ScrollOptions {
    var left: Double?
        get() = definedExternally
        set(value) = definedExternally
    var top: Double?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollToOptions(left: Double?, top: Double?, behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollToOptions {
    val o = js("({})")
    o["left"] = left
    o["top"] = top
    o["behavior"] = behavior
    o["return" + 0] = undefined
    return o
}

/**
 * Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
    open val media: String
    open val matches: Boolean
    open var onchange: ((Event) -> dynamic)?

    fun addListener(listener: EventListener?)
    fun addListener(listener: ((Event) -> Unit)?)
    fun removeListener(listener: EventListener?)
    fun removeListener(listener: (Event -> Unit)?)
}

/**
 * Exposes the JavaScript [MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin
 */
public external class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event

fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external open class Audio(src: String = definedExternally) : HTMLAudioElement {
override var onabort: ((Event) -> dynamic)?
override var onblur: ((FocusEvent) -> dynamic)?
override var oncancel: ((Event) -> dynamic)?
override var oncanplay: ((Event) -> dynamic)?
override var oncanplaythrough: ((Event) -> dynamic)?
override var onchange: ((Event) -> dynamic)?
override var onclick: ((MouseEvent) -> dynamic)?
override var onclose: ((Event) -> dynamic)?
override var oncontextmenu: ((MouseEvent) -> dynamic)?
override var oncuechange: ((Event) -> dynamic)?
override var onondblclick: ((MouseEventArgs) -> dynamic)?
override var onondrag: ((DragEvent) -> dynamic)?
override var onondragend: ((DragEvent) -> dynamic)?
override var onondragenter: ((DragEvent) -> dynamic)?
override var onondragexit: ((DragEvent) -> dynamic)?
override var onondragleave: ((DragEvent) -> dynamic)?
override var onondragover: ((DragEvent) -> dynamic)?
override var onondragstart: ((DragEvent) -> dynamic)?
override var onondrop: ((DragEvent) -> dynamic)?
override var ononended: ((Event) -> dynamic)?
override var ononerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
override var ononfocus: ((FocusEvent) -> dynamic)?
override var ononinput: ((InputEvent) -> dynamic)?
override var ononinvalid: ((Event) -> dynamic)?
override var ononkeydown: ((KeyboardEventArgs) -> dynamic)?
override var ononkeypress: ((KeyboardEventArgs) -> dynamic)?
override var ononkeyup: ((KeyboardEventArgs) -> dynamic)?
override var ononload: ((Event) -> dynamic)?
override var ononloadend: ((Event) -> dynamic)?
override var ononloadstart: ((Event) -> dynamic)?
override var ononreadystatechange: ((Event) -> dynamic)?
override var ononprogress: ((ProgressEvent) -> dynamic)?
override var ononratechange: ((Event) -> dynamic)?
override var ononreset: ((Event) -> dynamic)?
override var ononresize: ((Event) -> dynamic)?
override var ononscroll: ((Event) -> dynamic)?
override var ononseeked: ((Event) -> dynamic)?
override var ononseeking: ((Event) -> dynamic)?
override var ononselect: ((Event) -> dynamic)?
override var ononshow: ((Event) -> dynamic)?
override var ononstalled: ((Event) -> dynamic)?
override var ononsubmit: ((Event) -> dynamic)?
override var ononunpause: ((Event) -> dynamic)?
override var ononupdate: ((Event) -> dynamic)?
override var onontoggle: ((Event) -> dynamic)?
override var ononvolumechange: ((Event) -> dynamic)?
override var ononwaiting: ((Event) -> dynamic)?
override var ononpointercapture: ((PointerEventArgs) -> dynamic)?
override var ononpointercapture: ((PointerEventArgs) -> dynamic)?
override var ononpointercancel: ((PointerEventArgs) -> dynamic)?
override var ononpointerdown: ((PointerEventArgs) -> dynamic)?
override var ononpointerup: ((PointerEventArgs) -> dynamic)?
override var ononpointermove: ((PointerEventArgs) -> dynamic)
CanvasImageSource

public external interface CanvasImageSource : ImageBitmapSource

public external interface ImageBitmapSource

public external interface UnionMessagePortOrWindowProxy

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface DocumentReadyState {
    companion object
}


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanPlayTypeResult {
    companion object
}

public inline val CanPlayTypeResult.Companion.EMPTY: CanPlayTypeResult get() = "".asDynamic().unsafeCast<CanPlayTypeResult>()

public inline val CanPlayTypeResult.Companion.MAYBE: CanPlayTypeResult get() = "maybe".asDynamic().unsafeCast<CanPlayTypeResult>()

public inline val CanPlayTypeResult.Companion.PROBABLY: CanPlayTypeResult get() = "probably".asDynamic().unsafeCast<CanPlayTypeResult>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface TextTrackMode {
    companion object
}

public inline val TextTrackMode.Companion.DISABLED: TextTrackMode get() = "disabled".asDynamic().unsafeCast<TextTrackMode>()

public inline val TextTrackMode.Companion.HIDDEN: TextTrackMode get() = "hidden".asDynamic().unsafeCast<TextTrackMode>()

public inline val TextTrackMode.Companion.SHOWING: TextTrackMode get() = "showing".asDynamic().unsafeCast<TextTrackMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface TextTrackKind {
    companion object
}

public inline val TextTrackKind.Companion.SUBTITLES: TextTrackKind get() = "subtitles".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.CAPTIONS: TextTrackKind get() = "captions".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.DESCRIPTIONS: TextTrackKind get() = "descriptions".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.CHAPTERS: TextTrackKind get() = "chapters".asDynamic().unsafeCast<TextTrackKind>()

public inline val TextTrackKind.Companion.METADATA: TextTrackKind get() = "metadata".asDynamic().unsafeCast<TextTrackKind>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SelectionMode {
    companion object
}

public inline val SelectionMode.Companion.SELECT: SelectionMode get() = "select".asDynamic().unsafeCast<SelectionMode>()

public inline val SelectionMode.Companion.START: SelectionMode get() = "start".asDynamic().unsafeCast<SelectionMode>()

public inline val SelectionMode.Companion.END: SelectionMode get() = "end".asDynamic().unsafeCast<SelectionMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasFillRule {
    companion object
}

public inline val CanvasFillRule.Companion.NONZERO: CanvasFillRule get() = "nonzero".asDynamic().unsafeCast<CanvasFillRule>()

public inline val CanvasFillRule.Companion.EVENODD: CanvasFillRule get() = "evenodd".asDynamic().unsafeCast<CanvasFillRule>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface
"auto".asDynamic().unsafeCast<ScrollRestoration>()

public inline val ScrollRestoration.Companion.MANUAL: ScrollRestoration get() =
"manual".asDynamic().unsafeCast<ScrollRestoration>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ImageOrientation
{
    companion object
}

"none".asDynamic().unsafeCast<ScrollRestoration>()

"flipY".asDynamic().unsafeCast<ScrollRestoration>()

PremultiplyAlpha {
    companion object
}

public inline val PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() =
"none".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() =
"premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() =
"default".asDynamic().unsafeCast<PremultiplyAlpha>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ImageOrientation
{
    companion object
}

"none".asDynamic().unsafeCast<ScrollRestoration>()

"flipY".asDynamic().unsafeCast<ScrollRestoration>()

PremultiplyAlpha {
    companion object
}

public inline val PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() =
"none".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() =
"premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()

public inline val PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() =
"default".asDynamic().unsafeCast<PremultiplyAlpha>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ColorSpaceConversion
{
    companion object
}

public inline val ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() =
"none".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.PIXELATED: ColorSpaceConversion.get() =
"pixelated".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.LOW: ColorSpaceConversion get() =
"low".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.MEDIUM: ColorSpaceConversion.get() =
"medium".asDynamic().unsafeCast<ColorSpaceConversion>()

public inline val ColorSpaceConversion.Companion.HIGH: ColorSpaceConversion get() =
"high".asDynamic().unsafeCast<ColorSpaceConversion>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResizeQuality
{
    companion object
}

public inline val ResizeQuality.Companion.PIXELATED: ResizeQuality get() =
"pixelated".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.LOW: ResizeQuality get() =
"low".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.MEDIUM: ResizeQuality get() =
"medium".asDynamic().unsafeCast<ResizeQuality>()

public inline val ResizeQuality.Companion.HIGH: ResizeQuality get() =
"high".asDynamic().unsafeCast<ResizeQuality>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface BinaryType
{
    companion object
}

public inline val BinaryType.Companion.BLOB: BinaryType get() =
"blob".asDynamic().unsafeCast<BinaryType>()

public inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() =
"arraybuffer".asDynamic().unsafeCast<BinaryType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface WorkerType
{
    companion object
}

public inline val WorkerType.Companion.CLASSIC: WorkerType get() =
"classic".asDynamic().unsafeCast<WorkerType>()

public inline val WorkerType.Companion.MODULE: WorkerType get() =
"module".asDynamic().unsafeCast<WorkerType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ShadowRootMode
{
    companion object
}

public inline val ShadowRootMode.Companion.OPEN: ShadowRootMode get() =
"open".asDynamic().unsafeCast<ShadowRootMode>()

public inline val ShadowRootMode.Companion.CLOSED: ShadowRootMode get() =
"closed".asDynamic().unsafeCast<ShadowRootMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollBehavior
{
    companion object
}

public inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() =
"auto".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() =
"instant".asDynamic().unsafeCast<ScrollBehavior>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollLogicalPosition
{
    companion object
}

public inline val ScrollLogicalPosition.Companion.START: ScrollLogicalPosition get() =
"start".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() =
"end".asDynamic().unsafeCast<ScrollLogicalPosition>()
Open Source Used In DNAC Platform Arica 12351

definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["method"] = method
    o["headers"] = headers
    o["body"] = body
    o["referrer"] = referrer
    o["referrerPolicy"] = referrerPolicy
    o["mode"] = mode
    o["credentials"] = credentials
    o["cache"] = cache
    o["redirect"] = redirect
    o["integrity"] = integrity
    o["keepalive"] = keepalive
    o["window"] = window
    return o
}

/**
 * Exposes the JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to Kotlin
 */
public external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body {
    open val type: ResponseType
    open val url: String
    open val redirected: Boolean
    open val status: Short
    open val ok: Boolean
    open val statusText: String
    open val headers: Headers
    open val body: dynamic
    open val trailer: Promise<Headers>
    override val bodyUsed: Boolean
    override fun clone(): Response
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun json(): Promise<Any>
    override fun text(): Promise<String>

    companion object {
        fun error(): Response
        fun redirect(url: String, status: Short = definedExternally): Response
    }
}

public external interface ResponseInit {
    var status: Short? /* = 200 */
        get() = definedExternally
        set(value) = definedExternally
    var statusText: String? /* = "OK" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["status"] = status
    o["statusText"] = statusText
    o["headers"] = headers
    o["return"] = undefined
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestType {
    companion object
}

public inline val RequestType.Companion.EMPTY: RequestType get() = "".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.AUDIO: RequestType get() = "audio".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.FONT: RequestType get() = "font".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.IMAGE: RequestType get() = "image".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.SCRIPT: RequestType get() = "script".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.STYLE: RequestType get() = "style".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.TRACK: RequestType get() = "track".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.VIDEO: RequestType get() = "video".asDynamic().unsafeCast<RequestType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestDestination {
    companion object
}

public inline val RequestDestination.Companion.EMPTY: RequestDestination get() = "".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.DOCUMENT: RequestDestination get() = "document".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.EMBED: RequestDestination get() = "embed".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.FONT: RequestDestination get() = "font".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.IMAGE: RequestDestination get() = "image".asDynamic().unsafeCast<RequestDestination>()
"manifest".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.MEDIA: RequestDestination get() = "media".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.OBJECT: RequestDestination get() = "object".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.REPORT: RequestDestination get() = "report".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SCRIPT: RequestDestination get() = "script".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SERVICEWORKER: RequestDestination get() = "serviceworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SHAREDWORKER: RequestDestination get() = "sharedworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.STYLE: RequestDestination get() = "style".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.WORKER: RequestDestination get() = "worker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.XSLT: RequestDestination get() = "xslt".asDynamic().unsafeCast<RequestDestination>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestMode {
    companion object
}

public inline val RequestMode.Companion.NAVIGATE: RequestMode get() = "navigate".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.SAME_ORIGIN: RequestMode get() = "same-origin".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.NO_CORS: RequestMode get() = "no-cors".asDynamic().unsafeCast<RequestMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCredentials {
    companion object
}

public inline val RequestCredentials.Companion.OMIT: RequestCredentials get() = "omit".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.SAME_ORIGIN: RequestCredentials get() = "same-origin".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.INCLUDE: RequestCredentials get() = "include".asDynamic().unsafeCast<RequestCredentials>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCache {
    companion object
}

public inline val RequestCache.Companion.DEFAULT: RequestCache get() = "default".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_STORE: RequestCache get() = "no-store".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.RELOAD: RequestCache get() = "reload".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_CACHE: RequestCache get() = "no-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.FORCE_CACHE: RequestCache get() = "force-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = "only-if-cached".asDynamic().unsafeCast<RequestCache>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestRedirect {
    companion object
}

public inline val RequestRedirect.Companion.FOLLOW: RequestRedirect get() = "follow".asDynamic().unsafeCast<RequestRedirect>()

public inline val RequestRedirect.Companion.ERROR:
RequestRedirect get() = "error".asDynamic().unsafeCast<RequestRedirect>()

public inline val RequestRedirect.Companion.MANUAL: RequestRedirect get() = "manual".asDynamic().unsafeCast<RequestRedirect>()

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResponseType

companion object

public inline val ResponseType.Companion.BASIC: ResponseType get() = "basic".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.CORS: ResponseType get() = "cors".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.DEFAULT: ResponseType get() = "default".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.ERROR: ResponseType get() = "error".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.OPAQUE: ResponseType get() = "opaque".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.OPAQUEREDIRECT: ResponseType get() = "opaqueredirect".asDynamic().unsafeCast<ResponseType>()

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.dom.mediacapture

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.* * Exposes the JavaScript

[MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin

```
public external open class MediaStream() : EventTarget {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}
```

```
/**
 * Exposes the JavaScript
 * [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin
 *
```

```
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open var muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open val readyState: MediaStreamTrackState
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop()
    fun getCapabilities(): MediaTrackCapabilities
    fun getConstraints(): MediaTrackConstraints
    fun applyConstraints(constraints: MediaTrackConstraints = definedExternally): Promise<Unit>
}
```

```
/**
 * Exposes the JavaScript
 * [MediaTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSupportedConstraints) to Kotlin
 *
```

```
public external interface MediaTrackSupportedConstraints {
    var width: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var height: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var aspectRatio: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var frameRate: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var facingMode: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var resizeMode: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var volume: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var sampleRate: Boolean? = true
        get() = definedExternally
        set(value) = definedExternally
        var sampleSize: Boolean? = true
}
Open Source Used In DNAC Platform Arica 12354

```kotlin
definedExternally
set(value) = definedExternally
var echoCancellation: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var autoGainControl: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var noiseSuppression: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var latency: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var channelCount: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var deviceId: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally
var groupId: Boolean? /* = true */ get() = definedExternally
set(value) = definedExternally

set(value) = definedExternally

set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})") as MediaTrackSupportedConstraints
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceID"] = deviceID
    o["groupId"] = groupId
    return o
}

@kotlin.internal.InlineOnly
    val o = js("({})") as MediaTrackCapabilities
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceID"] = deviceID
    o["groupId"] = groupId
    return o
}
```

[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin

*/

public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced: Array<MediaTrackConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: boolean
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: boolean
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: boolean
        get() = definedExternally
        set(value) = definedExternally
    var latency: Int?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    return this
}

public external interface MediaTrackConstraintSet {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: boolean
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: boolean
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: boolean
        get() = definedExternally
        set(value) = definedExternally
    var latency: Int?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    return this
}

/**
 * Exposes the JavaScript [MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: boolean
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: boolean
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: boolean
        get() = definedExternally
        set(value) = definedExternally
    var latency: Int?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    return this
}

[MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin

* Exposes the JavaScript [MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin

*/

public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: boolean
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: boolean
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: boolean
        get() = definedExternally
        set(value) = definedExternally
    var latency: Int?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    return this
}

[MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin

* Exposes the JavaScript [MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin

*/

public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: boolean
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: boolean
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: boolean
        get() = definedExternally
        set(value) = definedExternally
    var latency: Int?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    return this
}
JavaScript [MediaStreamConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaStreamConstraints) to Kotlin

```kotlin
public external interface MediaStreamConstraints {
    var video: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var audio: dynamic /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
    val o = js("({})")
    o["video"] = video
    o["audio"] = audio
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

Kotlin

```kotlin
public external interface ConstrainablePattern {
    var onoverconstrained: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    fun getCapabilities(): Capabilities
    fun getConstraints(): Constraints
    fun applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
/**
 * Exposes the JavaScript [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin
 */
public external interface DoubleRange {
    var max: Double?
        get() = definedExternally
        set(value) = definedExternally
    var min: Double?
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
    val o = js("({})")
    o["max"] = max
    o["min"] = min
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface ConstrainDoubleRange : DoubleRange {
    var exact: Double?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Double?
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    o["max"] = max
    o["min"] = min
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface ULongRange {
    var max: Int?
        get() = definedExternally
        set(value) = definedExternally
    var min: Int?
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
public inline fun ULongRange(max: Int? = undefined, min: Int? = undefined): ULongRange {
    val o = js("({})")
    o["max"] = max
    o["min"] = min
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface ConstrainULongRange : ULongRange {
    var exact: Int?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Int?
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    o["max"] = max
    o["min"] = min
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
/**
 * Exposes the JavaScript [ConstrainBooleanParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainBooleanParameters) to Kotlin
 */
public external interface ConstrainBooleanParameters {
    var exact: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Boolean?
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
/**
 * Exposes the JavaScript [ConstrainDOMStringParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainDOMStringParameters) to Kotlin
 */
public external interface ConstrainDOMStringParameters {
    var exact: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var ideal: dynamic
        get() = definedExternally
        set(value) = definedExternally
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
@kotlin.internal.InlineOnly
public inline fun ConstrainDOMStringParameters(exact: dynamic = undefined, ideal: dynamic = undefined): ConstrainDOMStringParameters {
    val o = js("({})")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface Capabilities

@kotlin.internal.InlineOnly
public inline fun Capabilities(): Capabilities {
    val o = js("({})")
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface Settings

@kotlin.internal.InlineOnly
public inline fun Settings(): Settings {
    val o = js("({})")
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
public external interface ConstraintSet

@kotlin.internal.InlineOnly
public inline fun ConstraintSet(): ConstraintSet {
    val o = js("({})")
    return o
}
```

Open Source Used In DNAC Platform Arica 12357

```kotlin
/**
 * Exposes the JavaScript [ConstrainDoubleRange](https://developer.mozilla.org/en/docs/Web/API/ConstrainBooleanParameters) to Kotlin
 */
public external interface ConstrainDoubleRange {
    var exact: Double?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Double?
        get() = definedExternally
        set(value) = definedExternally
}
```
var advanced: Array<ConstraintSet>? = undefined
        Constraints {
            val o = js("({})")
            o["advanced"] = advanced
        } return o
}

@kotlin.internal.InlineOnly
public inline fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints {
    val o = js("({})")
    o["advanced"] = advanced
return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaStreamTrackState {
    companion object
}

public inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState get() = \"live\".asDynamic().unsafeCast<MediaStreamTrackState>()

public inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState get() = \"ended\".asDynamic().unsafeCast<MediaStreamTrackState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoFacingModeEnum {
    companion object
}

public inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum get() = \"user\".asDynamic().unsafeCast<VideoFacingModeEnum>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoResizeModeEnum {
    companion object
}

public inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = \"none\".asDynamic().unsafeCast<VideoResizeModeEnum>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = \"audioinput\".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = \"audiooutput\".asDynamic().unsafeCast<MediaDeviceKind>()
get() = definedExternally
    set(value) = definedExternally

    var isPrimary: Boolean? /* = false */

    get() = definedExternally
        set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun
    PointerEventInit(pointerId: Int? = 0, width: Double? = 1.0, height: Double? = 1.0, pressure: Float? = 0f,
        val o = js("({})")
        o["pointerId"] = pointerId
        o["width"] = width
        o["height"] = height
        o["pressure"] = pressure
        o["tangentialPressure"] = tangentialPressure
        o["tiltX"] = tiltX
        o["tiltY"] = tiltY
        o["twist"] = twist
        o["pointerType"] = pointerType
        o["isPrimary"] = isPrimary
        o["screenX"] = screenX
        o["screenY"] = screenY
        o["clientX"] = clientX
        o["clientY"] = clientY
        o["button"] = button
        o["buttons"] = buttons
        o["relatedTarget"] = relatedTarget
        o["region"] = region
        o["ctrlKey"] = ctrlKey
        o["shiftKey"] = shiftKey
        o["altKey"] = altKey
        o["metaKey"] = metaKey
        o["modifierAltGraph"] = modifierAltGraph
        o["modifierCapsLock"] = modifierCapsLock
        o["modifierFn"] = modifierFn
        o["modifierFnLock"] = modifierFnLock
        o["modifierHyper"] = modifierHyper
        o["modifierNumLock"] = modifierNumLock
        o["modifierScrollLock"] = modifierScrollLock
        o["modifierSymbol"] = modifierSymbol
        o["modifierSymbolLock"] = modifierSymbolLock
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    /**
     * Exposes the JavaScript [PointerEvent](https://developer.mozilla.org/en/docs/Web/API/PointerEvent) to Kotlin
     */
    public external open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally) : MouseEvent {
        open val
            pointerId: Int
        open val
            width: Double
        open val
            height: Double
        open val
            pressure: Float
        open val
            tangentialPressure: Float
        open val
            tiltX: Int
        open val
            tiltY: Int
        open val
            twist: Int
        open val
            pointerType: String
        open val
            isPrimary: Boolean

        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }
    }
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

var fill: Boolean? /* = true */
get() = definedExternally
set(value) = definedExternally

var stroke: Boolean? /* = false */
get() = definedExternally
set(value) = definedExternally

var markers: Boolean? /* = false */
get() = definedExternally
set(value) = definedExternally

var clipped: Boolean? /* = false */
get() = definedExternally
set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["fill"] = fill
    o["stroke"] = stroke
    o["markers"] = markers
    o["clipped"] = clipped
    return o
}

/**
 * Exposes the JavaScript
 */
public external abstract class SVGGraphicsElement : SVGElement, SVGTests {
    open val transform: SVGAnimatedTransformList
    fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect
    fun getCTM(): DOMMatrix?
    fun getScreenCTM(): DOMMatrix?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

    /**
     * Exposes the JavaScript
     */
    public external abstract class SVGGeometryElement : SVGGraphicsElement {
        open val pathLength: SVGAnimatedNumber
        fun isPointInFill(point: DOMPoint): Boolean
        fun isPointInStroke(point: DOMPoint): Boolean
        fun getTotalLength(): Float
        fun getPointAtLength(distance: Float): DOMPoint

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
            val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
        }
    }

    /**
     * Exposes the JavaScript
     */
    public external abstract class SVGNumber {
        open var value: Float
    }

    /**
     * Exposes the JavaScript
     */
    public external abstract class SVGLength {
        open val unitType: Short
        open var value: Float
        open var valueInSpecifiedUnits: Float
        fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
        fun convertToSpecifiedUnits(unitType: Short)

        companion object {
            val SVG_LENGTHTYPE_UNKNOWN: Short
            val SVG_LENGTHTYPE_NUMBER: Short
            val SVG_LENGTHTYPE_PERCENTAGE: Short
            val SVG_LENGTHTYPE_EMS: Short
            val SVG_LENGTHTYPE_EXS: Short
            val SVG_LENGTHTYPE_PX: Short
            val SVG_LENGTHTYPE_CM: Short
            val SVG_LENGTHTYPE_MM: Short
            val SVG_LENGTHTYPE_IN: Short
            val SVG_LENGTHTYPE_PT: Short
            val SVG_LENGTHTYPE_PC:
Exposes the JavaScript `SVGAngle` to Kotlin:

```kotlin
class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)
    companion object {
        val SVG_ANGLETYPE_UNKNOWN: Short
        val SVG_ANGLETYPE_UNSPECIFIED: Short
        val SVG_ANGLETYPE_DEG: Short
        val SVG_ANGLETYPE_RAD: Short
        val SVG_ANGLETYPE_GRAD: Short
    }
}
```

Exposes the JavaScript `SVGLengthList` to Kotlin:

```kotlin
class SVGLengthList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGLength)
    fun insertItemBefore(newItem: SVGLength, index: Int)
    fun replaceItem(newItem: SVGLength, index: Int)
    fun removeItem(index: Int)
    fun appendItem(newItem: SVGLength)
    fun getItem(index: Int)
}
```

Exposes the JavaScript `SVGNumberList` to Kotlin:

```kotlin
class SVGNumberList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGNumber)
    fun insertItemBefore(newItem: SVGNumber, index: Int)
    fun replaceItem(newItem: SVGNumber, index: Int)
    fun removeItem(index: Int)
    fun appendItem(newItem: SVGNumber)
    fun getItem(index: Int)
}
```

Exposes the JavaScript `SVGAnimatedBoolean` to Kotlin:

```kotlin
class SVGAnimatedBoolean {
    open var baseVal: Boolean
    open val animVal: Boolean
}
```

Exposes the JavaScript `SVGAnimatedEnumeration` to Kotlin:

```kotlin
class SVGAnimatedEnumeration {
    open var baseVal: Short
    open val animVal: Short
}
```

Exposes the JavaScript `SVGAnimatedInteger` to Kotlin:

```kotlin
class SVGAnimatedInteger {
    open var baseVal: Int
    open val animVal: Int
}
```

Exposes the JavaScript `SVGAnimatedNumber` to Kotlin:

```kotlin
class SVGAnimatedNumber {
    open var baseVal: Float
    open val animVal: Float
}
```

Exposes the JavaScript `SVGAnimatedLength` to Kotlin:

```kotlin
class SVGAnimatedLength {
    open val baseVal: SVGLength
    open val animVal: SVGLength
}
```

Exposes the JavaScript `SVGAnimatedAngle` to Kotlin:

```kotlin
class SVGAnimatedAngle {
    open val baseVal: SVGAngle
    open val animVal: SVGAngle
}
```
SVGAnimatedString

```kotlin
public external abstract class SVGAnimatedString {
    open var baseVal: String
    open val animVal: String
}

/**
 * Exposes the JavaScript [SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect) to Kotlin
 */
public external abstract class SVGAnimatedRect {
    open val baseVal: DOMRect
    open val animVal: DOMRectReadOnly
}

/**
 * Exposes the JavaScript [SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin
 */
public external abstract class SVGAnimatedNumberList {
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberList
}

/**
 * Exposes the JavaScript [SVGAnimatedLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLengthList) to Kotlin
 */
public external abstract class SVGAnimatedLengthList {
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthList
}

/**
 * Exposes the JavaScript [SVGStringList](https://developer.mozilla.org/en/docs/Web/API/SVGStringList) to Kotlin
 */
public abstract class SVGStringList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: String): String
    fun insertItemBefore(newItem: String, index: Int): String
    fun replaceItem(newItem: String, index: Int): String
    fun removeItem(index: Int): String
    fun appendItem(newItem: String): String
    fun getItem(index: Int): String
}

@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.get(index: Int): String? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.set(index: Int, newItem: String) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGUnitTypes](https://developer.mozilla.org/en/docs/Web/API/SVGUnitTypes) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGUnitTypes {
    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    }
}

/**
 * Exposes the JavaScript [SVGTests](https://developer.mozilla.org/en/docs/Web/API/SVGTests) to Kotlin
 */
public external interface SVGTests {
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
}

public external interface SVGFitToViewBox {
    val viewBox: SVGAnimatedRect
    val preserveAspectRatio: SVGAnimatedPreserveAspectRatio
}

/**
 * Exposes the JavaScript [SVGZoomAndPan](https://developer.mozilla.org/en/docs/Web/API/SVGZoomAndPan) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface SVGZoomAndPan {
    var zoomAndPan: Short

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
    }
}

/**
 * Exposes the JavaScript [SVGURIReference](https://developer.mozilla.org/en/docs/Web/API/SVGURIReference) to Kotlin
 */
public external interface SVGURIReference {
    val href: SVGAnimatedString
}

/**
 * Exposes the JavaScript [SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin
 */
public external abstract class SVGSVGElement : SVGGraphicsElement, SVGFitToViewBox, SVGZoomAndPan, WindowEventHandlers {
    open var x: SVGAnimatedLength
    open var y: SVGAnimatedLength
    open var width: SVGAnimatedLength
    open var height: SVGAnimatedLength
    open var currentScale: Float
    open var currentTranslate: DOMPointReadOnly
    fun getIntersectionList(rect: DOMRectReadOnly): NodeList
    fun getEnclosureList(rect: DOMRectReadOnly): NodeList
    fun checkIntersection(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun checkEnclosure(element: SVGElement, rect: DOMRectReadOnly): Boolean
    fun deselectAll(): Void
    fun createSVGNumber(): SVGNumber
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGMatrix(): DOMMatrix
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
```
getElementById(elementId: String): Element
fun suspendRedraw(maxWaitMilliseconds: Int): Int
fun unsuspendRedraw(suspendHandleID: Int)
fun unsuspendRedrawAll()
fun forceRedraw()

companion object {
val SVG_ZOOMANDPAN_UNKNOWN: Short
val SVG_ZOOMANDPAN_DISABLE: Short
val SVG_ZOOMANDPAN_MAGNIFY: Short
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

/**
 * Exposes the JavaScript [SVGGElement](https://developer.mozilla.org/en/docs/Web/API/SVGGElement) to Kotlin
 */
public external abstract class SVGGElement : SVGGraphicsElement{
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGUnknownElement](https://developer.mozilla.org/en/docs/Web/API/SVGUnknownElement) to Kotlin
 */
public external abstract class SVGUnknownElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin
 */
public external abstract class SVGDefsElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGDescElement](https://developer.mozilla.org/en/docs/Web/API/SVGDescElement) to Kotlin
 */
public external abstract class SVGDescElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}
}

/**
 * Exposes the JavaScript [SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin
 */
public external abstract class SVGMetadataElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTitleElement](https://developer.mozilla.org/en/docs/Web/API/SVGTitleElement) to Kotlin
 */
public external abstract class SVGTitleElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin
 */
public external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin
 */
public external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val instanceRoot: SVGElement?
    open val animatedInstanceRoot: SVGElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
Open Source Used In DNAC Platform Arica 12365
setSkewY(angle: Float)/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun clear()/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun initialize(newItem: SVGTransform)/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun removeItem(index: Int): SVGTransform/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun appendItem(newItem: SVGTransform): SVGTransform/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun consolidate(): SVGTransform?/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
fun getItem(index: Int): SVGTransform

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.get(index: Int): SVGTransform? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
 */
public external abstract class SVGPathElement : SVGGeometryElement
val y: SVGAnimatedLength open val width: SVGAnimatedLength open val height: SVGAnimatedLength
open val rx: SVGAnimatedLength open val ry: SVGAnimatedLength

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin
 */
public external abstract class SVGCircleElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin
 */
public external abstract class SVGEllipseElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val rx: SVGAnimatedLength
    open val ry: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin
 */
public external abstract class SVGLineElement : SVGGeometryElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
 */
public external abstract class SVGPathElement : SVGGeometryElement {
    open val d: SVGPathData

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
 */
public external abstract class SVGRectElement : SVGGeometryElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
 */
public external abstract class SVGRectElement : SVGGeometryElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
 */
public external abstract class SVGRectElement : SVGGeometryElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
 */
public external abstract class SVGMeshElement : SVGMixedElement {
    open val count: Int
    open val at: Int
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
 */
public external abstract class SVGMeshElement : SVGMixedElement {
    open val count: Int
    open val at: Int
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin
 */
public external abstract class SVGMeshElement : SVGMixedElement {
    open val count: Int
    open val at: Int
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin
 */
public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
    fun removeItem(index: Int): DOMPoint
    fun appendItem(newItem: DOMPoint): DOMPoint
    fun getItem(index: Int): DOMPoint
}

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript
 [SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin
 */
public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin
 */
public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextContentElement) to Kotlin
 */
public external abstract class SVGTextContentElement : SVGGraphicsElement {
    open val textLength: SVGAnimatedLength
    open val lengthAdjust: SVGAnimatedEnumeration
    fun getNumberOfChars(): Int
    fun getComputedTextLength(): Float
    fun getSubStringLength(charnum: Int, nchars: Int): Float
    fun getStartPositionOfChar(charnum: Int): DOMPoint
    fun getEndPositionOfChar(charnum: Int): DOMPoint
    fun getExtentOfChar(charnum: Int): DOMRect
    fun getRotationOfChar(charnum: Int): Float
}
public external abstract class SVGTextPositioningElement : SVGTextContentElement {
    open val x: SVGAnimatedLengthList
    open val y: SVGAnimatedLengthList
    open val dx: SVGAnimatedLengthList
    open val dy: SVGAnimatedLengthList
    open val rotate: SVGAnimatedNumberList
}

/**
 * Exposes the JavaScript [SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin
 */
public external abstract class SVGTextElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGTSpanElement](https://developer.mozilla.org/en/docs/Web/API/SVGTSpanElement) to Kotlin
 */
public external abstract class SVGTSpanElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin
 */
public external abstract class SVGTextPathElement : SVGTextContentElement, SVGURIReference {
    open val startOffset: SVGAnimatedLength
    open val method: SVGAnimatedEnumeration
    open val spacing: SVGAnimatedEnumeration

    companion object {
        val TEXTPATH_METHODTYPE_UNKNOWN: Short
        val TEXTPATH_METHODTYPE_ALIGN: Short
        val TEXTPATH_METHODTYPE_STRETCH: Short
        val TEXTPATH_SPACINGTYPE_UNKNOWN: Short
        val TEXTPATH_SPACINGTYPE_AUTO: Short
        val TEXTPATH_SPACINGTYPE_EXACT: Short
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGImageElement](https://developer.mozilla.org/en/docs/Web/API/SVGImageElement) to Kotlin
 */
public external abstract class SVGImageElement : SVGGraphicsElement, SVGURIReference, HTMLOrSVGImageElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open var crossOrigin: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGForeignObjectElement](https://developer.mozilla.org/en/docs/Web/API/SVGForeignObjectElement) to Kotlin
 */
public external abstract class SVGForeignObjectElement : SVGGraphicsElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
**Open Source Used In DNAC Platform Arica**

```kotlin

public external abstract class SVGMarkerElement : SVGElement, SVGFitToViewBox {
    open val refX: SVGAnimatedLength
    open val refY: SVGAnimatedLength
    open val markerUnits: SVGAnimatedEnumeration
    open val markerWidth: SVGAnimatedLength
    open val markerHeight: SVGAnimatedLength
    open val orientType: SVGAnimatedEnumeration
    open val orientAngle: SVGAnimatedAngle
    fun setOrientToAuto()
    fun setOrientToAngle(angle: SVGAngle)

    companion object {
        val SVG_MARKERUNITS_UNKNOWN: Short
        val SVG_MARKERUNITS_USERSPACEONUSE: Short
        val SVG_MARKERUNITS_STROKEWIDTH: Short
        val SVG_MARKER_ORIENT_UNKNOWN: Short
        val SVG_MARKER_ORIENT_AUTO: Short
        val SVG_MARKER_ORIENT_ANGLE: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGSolidcolorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement) to Kotlin
 */
public external abstract class SVGSolidcolorElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin
 */
public external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {
    open val gradientUnits: SVGAnimatedEnumeration
    open val gradientTransform: SVGAnimatedTransformList
    open val spreadMethod: SVGAnimatedEnumeration

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to
 */
```
Kotlin
*/
public external abstract class SVGLinearGradientElement : SVGGradientElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    }
}

/**
 * Exposes the JavaScript [SVGRadialGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGRadialGradientElement) to Kotlin
 */
public external abstract class SVGRadialGradientElement : SVGGradientElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength
    open val fx: SVGAnimatedLength
    open val fy: SVGAnimatedLength
    open val fr: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    }
}

public external abstract class SVGMeshGradientElement : SVGGradientElement {
    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
    }
}

public external abstract class SVGMeshrowElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
    }
}
var onload: ((Event) -> dynamic)?
var onabort: ((Event) -> dynamic)?
var onerror: ((Event) -> dynamic)?

var onloadend: ((Event) -> dynamic)?
fun readAsArrayBuffer(blob: Blob): ArrayBuffer
fun readAsBinaryString(blob: Blob): String
fun readAsDataURL(blob: Blob): String

fun abort()

companion object {
    val EMPTY: Short
    val LOADING: Short
    val DONE:
}

/**
 * Exposes the JavaScript
 [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}

/**
 * Exposes the JavaScript
 [Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin
 */
public external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    var onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: Array<out Int>
    open val timestamp: Number
    open val renotify: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun close()

    companion object {
        val permission: NotificationPermission
        val maxActions: Int
        fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>
    }
}

@kotlin.internal.InlineOnly

var onerror: ((Event) -> dynamic)?
var onabort: ((Event) -> dynamic)?
var onload: ((Event) -> dynamic)?
var onloadend: ((Event) -> dynamic)?
fun readAsArrayBuffer(blob: Blob): ArrayBuffer
fun readAsBinaryString(blob: Blob): String
fun readAsDataURL(blob: Blob): String
fun abort()

companion object {
    val EMPTY: Short
    val LOADING: Short
    val DONE:
}

/**
 * Exposes the JavaScript
 [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}

/**
 * Exposes the JavaScript
 [Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin
 */
public external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    var onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: dynamic
    open val timestamp: Number
    open val renotify: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun close()

    companion object {
        val permission: NotificationPermission
        val maxActions: Int
        fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>
    }
}

@kotlin.internal.InlineOnly

var onerror: ((Event) -> dynamic)?
var onabort: ((Event) -> dynamic)?
var onload: ((Event) -> dynamic)?
var onloadend: ((Event) -> dynamic)?
fun readAsArrayBuffer(blob: Blob): ArrayBuffer
fun readAsBinaryString(blob: Blob): String
fun readAsDataURL(blob: Blob): String
fun abort()

companion object {
    val EMPTY: Short
    val LOADING: Short
    val DONE:
}

/**
 * Exposes the JavaScript
 [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
    fun readAsText(blob: Blob, label: String = definedExternally): String
    fun readAsDataURL(blob: Blob): String
}

/**
 * Exposes the JavaScript
 [Notification](https://developer.mozilla.org/en/docs/Web/API/Notification) to Kotlin
 */
public external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    var onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: dynamic
    open val timestamp: Number
    open val renotify: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun close()

    companion object {
        val permission: NotificationPermission
        val maxActions: Int
        fun requestPermission(deprecatedCallback: (NotificationPermission) -> Unit = definedExternally): Promise<NotificationPermission>
    }
}

@kotlin.internal.InlineOnly

var onerror: ((Event) -> dynamic)?
var onabort: ((Event) -> dynamic)?
var onload: ((Event) -> dynamic)?
var onloadend: ((Event) -> dynamic)?
fun readAsArrayBuffer(blob: Blob): ArrayBuffer
fun readAsBinaryString(blob: Blob): String
fun readAsDataURL(blob: Blob): String
fun abort()
false, data: Any? = null, actions: Array<NotificationAction>? = arrayOf()): NotificationOptions {
    val o = js("({})")
    o["dir"] = dir
    o["lang"] = lang
    o["body"] = body
    o["tag"] = tag
    o["image"] = image
    o["icon"] = icon
    o["badge"] = badge
    o["sound"] = sound
    o["vibrate"] = vibrate
    o["timestamp"] = timestamp
    o["renotify"] = renotify
    o["silent"] = silent
    o["noscreen"] = noscreen
    o["requireInteraction"] = requireInteraction
    o["sticky"] = sticky
    o["data"] = data
    o["actions"] = actions
    return o
}

public external interface NotificationAction {
    var action: String?
        get() = definedExternally
        set(value) = definedExternally
    var title: String?
        get() = definedExternally
        set(value) = definedExternally
    var icon: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["action"] = action
    o["title"] = title
    o["icon"] = icon
    return o
}

public external interface GetNotificationOptions {
    var tag: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetNotificationOptions(tag: String? = ""): GetNotificationOptions {
    val o = js("({})")
    o["tag"] = tag
    return o
}

/**
 * Exposes the JavaScript [NotificationEvent](https://developer.mozilla.org/en/docs/Web/API/NotificationEvent) to Kotlin
 */
public external open class NotificationEvent(type: String, eventInitDict: NotificationEventInit) : ExtendableEvent {
    open val notification: Notification
    open val action: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface NotificationEventInit : ExtendableEventInit {
    var notification: Notification?
        get() = definedExternally
        set(value) = definedExternally
    var action: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var bubbles: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var composed: Boolean?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["notification"] = notification
    o["action"] = action
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don't implement this interface!
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationPermission {
    companion object
}

public inline val NotificationPermission.Companion.DEFAULT: NotificationPermission get() = "default".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.DENIED: NotificationPermission get() = "denied".asDynamic().unsafeCast<NotificationPermission>()

public inline val NotificationPermission.Companion.GRANTED: NotificationPermission get() = "granted".asDynamic().unsafeCast<NotificationPermission>()

/* please, don't implement this interface!
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NotificationDirection {
    companion object
}

public inline val NotificationDirection.Companion.AUTO: NotificationDirection get() = "auto".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.LTR: NotificationDirection get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.CompanionRTL: NotificationDirection get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()
org.w3c.notifications.*

import org.w3c.performance.*

import org.w3c.xhr.*

/**
 * Exposes the JavaScript
 * [ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin
 */
public external abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker,
UnionClientOrMessagePortOrServiceWorker {
    open val scriptURL: String
    open val state:
    ServiceWorkerState
    open var onstatechange: ((Event) -> dynamic)?
    fun postMessage(message: Any?,
transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript
 * [ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to
Kotlin
 */
public external abstract class ServiceWorkerRegistration : EventTarget {
    open val installing:
    ServiceWorker?
    open val waiting: ServiceWorker?
    open val active: ServiceWorker?
    open val scope:
    String
    open var onupdatefound: ((Event) -> dynamic)?
    open val APISpace: dynamic
    fun update():
Promise<Unit>
    fun unregister(): Promise<Boolean>
    fun showNotification(title: String, options:
NotificationOptions = definedExternally): Promise<Unit>
    fun getNotifications(filter: GetNotificationOptions =
definedExternally): Promise<Array<Notification>>
    fun methodName(): Promise<dynamic>
}

/**
 * Exposes the JavaScript
 * [ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin
 */
public external abstract class ServiceWorkerContainer : EventTarget {
    open val controller:
    ServiceWorker?
    open val ready: Promise<ServiceWorkerRegistration>
    open var oncontrollerchange:
    ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun register(scriptURL: String,
onoptions: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>
    fun getRegistration(clientURL: String = definedExternally): Promise<Any?>
    fun getRegistrations():
Promisedynamic
}

public external interface
RegistrationOptions {
    var scope: String?
        get() = definedExternally
        set(value) = definedExternally
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {
    val o = js("({})")
    o["scope"] = scope
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript
 * [ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to
Kotlin
 */
public external open class ServiceWorkerMessageEvent(type: String, eventInitDict:
ServiceWorkerMessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin:
    String
    open val lastEventId: String
    open val source: UnionMessagePortOrServiceWorker?
    open val ports: Array<MessagePort>?
    open val client:
    ServiceWorker?
    open var oninstall: ((Event) -> dynamic)?
    open var onactivate: ((Event) -> dynamic)?
    open var onfetch: ((FetchEvent) -> dynamic)?
    open var onsleep: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    open var onmessageerror: ((MessageEvent) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onstatechange: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
}

public external interface ServiceWorkerMessageEventInit : EventInit {
    var data: Any?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
    var lastEventId: String?
        get() = definedExternally
        set(value) = definedExternally
    var source: UnionMessagePortOrServiceWorker?
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ServiceWorkerMessageEventInit(data:
Any? = undefined, origin: String? = undefined, lastEventId: String? = undefined, source:
UnionMessagePortOrServiceWorker? = undefined, ports: Array<MessagePort>? = undefined, bubbles: Boolean? =
false, cancelable: Boolean? = false): ServiceWorkerMessageEventInit {
    val o = js("{"(1)"})
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] =
    source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 * [ServiceWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerGlobalScope) to
Kotlin
 */
public external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {
    open var clients: Array<
Clients>
    open var registration: ServiceWorkerRegistration
    open var oninstall: ((Event) -> dynamic)?
    open var onactivate: ((Event) -> dynamic)?
    open var onfetch: ((FetchEvent) -> dynamic)?
}
onforeignfetch: ((Event) -> dynamic)?

open var onmessage: ((MessageEvent) -> dynamic)?
open var onnotificationclick: ((NotificationEvent) -> dynamic)?
open var onnotificationclose: ((NotificationEvent) -> dynamic)?
open var onfunctionalevent: ((Event) -> dynamic)?
fun skipWaiting():

Promise<Unit?>

/**
 * Exposes the JavaScript
 */
public external abstract class Client :
UnionClientOrMessagePortOrServiceWorker {

open val url: String
open val frameType: FrameType
open val id: String
fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript
 */
public external abstract class WindowClient : Client {

open val visibilityState: dynamic
open val focused: Boolean
fun focus(): Promise<WindowClient>
fun navigate(url: String):
}

/**
 * Exposes the JavaScript
 */
public external abstract class Clients {

fun get(id: String): Promise<Any?>
fun matchAll(options: ClientQueryOptions = definedExternally):
fun claim(): Promise<Unit>
}

/**
 * Exposes the JavaScript
 */
public external interface ClientQueryOptions {

var includeUncontrolled: Boolean? /* = false */
var type: ClientType? /* = ClientType.WINDOW */
}

@kotlin.internal.InlineOnly
public inline fun ClientQueryOptions(includeUncontrolled: Boolean? = false, type: ClientType? = ClientType.WINDOW): ClientQueryOptions {

val o = js("({})")


/**
 * Exposes the JavaScript
 */
public external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {

fun waitUntil(f: Promise<Any?>)

companion object {

val NONE:
val CAPTURING_PHASE:
val AT_TARGET:
val BUBBLING_PHASE:
}

public external interface ExtendableEventInit : EventInit

@kotlin.internal.InlineOnly

val o = js("({})")


/**
 * Exposes the JavaScript
 */
public external open class InstallEvent(type: String, eventInitDict: InstallEventInit = definedExternally) : ExtendableEvent {

fun registerForeignFetch(options: ForeignFetchOptions)

companion object {

val NONE:
val CAPTURING_PHASE:
val AT_TARGET:
val BUBBLING_PHASE:
}

public external interface ForeignFetchOptions {

var scopes: Array<String>?
var origins: Array<String>?
}

@kotlin.internal.InlineOnly
public inline fun ForeignFetchOptions(scopes: Array<String>?, origins: Array<String>?): ForeignFetchOptions {

val o = js("({})")


/**
 * Exposes the JavaScript
 */
public external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {

open val clientId: String?
fun respondWith(r: Promise<Response>)

companion object {

val NONE:
val CAPTURING_PHASE:
val AT_TARGET:
val BUBBLING_PHASE:
}

public external interface FetchEventInit : EventInit

@kotlin.internal.InlineOnly

val o = js("({})")


Open Source Used In DNAC Platform Arica 12379
    val o = js("({})")
    o["overrideSearch"] = overrideSearch
    o["overrideMethod"] = overrideMethod
    o["overrideVary"] = overrideVary
    o["cacheName"] = cacheName
    return o
}

public external interface CacheBatchOperation {
    var type: String?
        get() = definedExternally
        set(value) = definedExternally
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var response: Response?
        get() = definedExternally
        set(value) = definedExternally
    var options: CacheQueryOptions?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["type"] = type
    o["request"] = request
    o["response"] = response
    o["options"] = options
    return o
}

/**
 * Exposes the JavaScript
 */
public external abstract class CacheStorage {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun has(cacheName: String): Promise<Boolean>
    fun open(cacheName: String): Promise<Cache>
    fun delete(cacheName: String): Promise<Boolean>
    fun keys(): Promise<Array<String>>
}

public external open class FunctionalEvent : ExtendableEvent {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface UnionMessagePortOrServiceWorker

public external interface UnionClientOrMessagePortOrServiceWorker

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ServiceWorkerState {
    companion object
}

public inline val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() = "installing".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() = "installed".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() = "activating".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() = "activated".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.REDUNDANT: ServiceWorkerState get() = "redundant".asDynamic().unsafeCast<ServiceWorkerState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface FrameType {
    companion object
}

public inline val FrameType.Companion.AUXILIARY: FrameType get() = "auxiliary".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.TOP_LEVEL: FrameType get() = "top-level".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NESTED: FrameType get() = "nested".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NONE: FrameType get() = "none".asDynamic().unsafeCast<FrameType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ClientType {
    companion object
}

public inline val ClientType.Companion.WINDOW: ClientType get() = "window".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.WORKER: ClientType get() = "worker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.SHAREDWORKER: ClientType get() = "sharedworker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.ALL: ClientType get() = "all".asDynamic().unsafeCast<ClientType>()

/* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details

// Open Source Used In DNAC Platform Arica 12381
open var responseType: XMLHttpRequestResponseType
open val responseText: String
fun overrideMimeType(mime: String)

companion object {
    val UNSENT: Short
    val OPENED: Short
    val HEADERS_RECEIVED: Short
    val LOADING: Short
    val DONE: Short
}

interface XMLHttpRequestResponseType {
    companion object
}

val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType = "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType = "arraybuffer".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")

interface XMLHttpRequestEventTarget {
    open var onreadystatechange: ((Event) -> dynamic)?
    open val readyState: Short
    var timeout: Int
    var withCredentials: Boolean
    open val upload: XMLHttpRequestUpload
    open val responseURL: String
    open val status: Short
    open val statusText: String
    var responseType: XMLHttpRequestResponseType
    open val response: Any?
    open val responseText: String
    open val responseXML: Document?
    fun open(method: String, url: String)
    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String?
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Short
        val OPENED: Short
        val HEADERS_RECEIVED: Short
        val LOADING: Short
        val DONE: Short
    }

    open val onloadstart: ((ProgressEvent) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onabort: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onload: ((Event) -> dynamic)?
    open var ontimeout: ((Event) -> dynamic)?
    open var onloadend: ((Event) -> dynamic)?
}

abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget

interface ProgressEventInit : EventInit {
    var lengthComputable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var loaded: Number? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var total: Number? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["lengthComputable"] = lengthComputable
    o["loaded"] = loaded
    o["total"] = total
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

fun XMLHttpRequestEventTarget.onload(event: Event) = event

XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = 
"blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = 
"document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = 
"json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.annotation.AnnotationRetention.BINARY
import kotlin.annotation.AnnotationRetention.SOURCE
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind
import kotlin.reflect.KClass

/**
 * Signals that the annotated annotation class is a marker of an experimental API.
 * Any declaration annotated with that marker is considered an experimental declaration
 * and its call sites should accept the experimental aspect of it either by using [UseExperimental],
 * or by being annotated with that marker themselves, effectively causing further propagation of that experimental aspect.
 * This class is experimental itself and can only be used with the compiler argument `-Xuse-experimental=kotlin.Experimental`.
 */
@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Suppress("ANNOTATION_CLASS_MEMBER")
public annotation class Experimental(val level: Level = Level.ERROR) {
    /**
     * Severity of the diagnostic that should be reported on usages of experimental API which did not explicitly accept the experimental aspect
     * of that API either by using [UseExperimental] or by being annotated with the corresponding marker annotation.
     */
    public enum class Level {
        /** Specifies that a warning should be reported on incorrect usages of this experimental API. */
        WARNING,
        /** Specifies that an error should be reported on incorrect usages of this experimental API. */
        ERROR,
    }
}

/**
 * Allows to use experimental API denoted by the given markers in the annotated file, declaration, or expression.
 * If a declaration is annotated with [UseExperimental], its usages are **not** required to opt-in to that experimental API.
 * This class is experimental itself and can only be used with the compiler argument `-Xuse-experimental=kotlin.Experimental`.
 */
@Target(
    CLASS,
    PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION,
    PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
    TYPEALIAS)
@Retention(SOURCE)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Suppress("ANNOTATION_CLASS_MEMBER")
public annotation class UseExperimental(
    vararg val markerClass: KClass<out Annotation>
)

@Target(CLASS, PROPERTY, CONSTRUCTOR, FUNCTION,
    PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,
    TYPEALIAS)
@Retention(SOURCE)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@public annotation class UseExperimental
(vararg val markers: KClass<out Annotation>)

/**
 * This annotation marks the standard library API that is considered experimental and is not subject to the general compatibility guarantees
 * given for the standard library: the behavior of such API may be changed or the API may be removed completely in any future release.
 * Beware using the annotated API especially if you're developing a library, since your library might become binary incompatible
 * with the future versions of the standard library.
 * Any usage of a declaration annotated with `@ExperimentalStdlibApi` must be accepted either by
 * annotating that usage with the [UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalStdlibApi::class)`.
 * or by using the compiler argument `-Xuse-experimental=kotlin.ExperimentalStdlibApi`.
 */
@Target(ANNOTATION_CLASS)
public annotation class ExperimentalStdlibApi:
(Experimental::class)

Open Source Used In DNAC Platform Arica

```kotlin
@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
public annotation class ExperimentalStdlibApi

@MustBeDocumented
@ExperimentalStdlibApi
@SinceKotlin("1.3") // TODO: Remove experimental status from itself and advance SinceKotlin to 1.4

public annotation class ExperimentalTypeInference

@package kotlin
@import kotlin.experimental.ExperimentalTypeInference

/**
 * Allows to infer generic type arguments of a function from the calls in the annotated function parameter of that function.
 * When this annotation is placed on a generic function parameter of a function, it enables to infer the type arguments of that generic function from the lambda body passed to that parameter.
 * The calls that affect inference are either members of the receiver type of an annotated function parameter or extensions for that type. The extensions must be themselves annotated with `@BuilderInference`.
 * Example: we declare a function `fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T>` and use it like `sequence { yield("result") }` and it is inferred from the argument of the `SequenceScope.yield` function, that is called inside the lambda passed to the `sequence` function.
 * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.
 */
@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
public annotation class BuilderInference

/**
 * The experimental multiplatform support API marker.
 *
 * Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by annotating that usage with the [UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalMultiplatform::class)`, or by using the compiler argument `-Xuse-experimental=kotlin.ExperimentalMultiplatform`.
 */
@Experimental
@MustBeDocumented
@Target(CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class ExperimentalMultiplatform

/**
 * Marks an expected annotation class that it isn’t required to have actual counterparts in all platforms.
 * This annotation is only applicable to `expect` annotation classes in multi-platform projects and marks that class as "optional".
 * Optional expected class is allowed to have no corresponding actual class on the platform. Optional annotations can only be used to annotate something, not as types in signatures. If an optional annotation has no corresponding actual class on a platform, the annotation entries where it’s used are simply erased when compiling code on that platform.
 * Note: this annotation is experimental, see [ExperimentalMultiplatform] on how to opt-in for it.
 */
@Target(ANNOTATION_CLASS)
@Retention(AnnotationRetention.BINARY)
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class OptionalExpectation
```

@package kotlin
@import kotlin.collections
@import kotlin.js

/**
 * Provides a skeletal implementation of the read-only [Collection] interface.
 * @param E the type of elements contained in the collection. The collection is covariant on its element type.
 */
@SinceKotlin("1.1")
public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }
    override fun containsAll(elements: Collection<E>): Boolean = elements.all { contains(it) } // use when js will support bound refs: elements.all(this::contains)
    override fun
isNotEmpty(): Boolean = size == 0

override fun toString(): String = joinToString("\", ",\"") { 
    it.toString()
} */

* Returns new array of type `Array<Any?>` with the elements of this collection.
* Returns protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)
* Fills the provided [array] or creates new array of the same type
* and fills it with the elements of this collection.
* protected open fun <T> toArray(array: Array<T>): Array<T> = copyToArrayImpl(this, array)
*/

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*

package kotlin.collections

private enum class State {
    Ready,
    Done,
    Failed
}

/**
 * A base class to simplify implementing iterators so that implementations only have to implement [computeNext] to * implement the iterator, calling [done] when the iteration is complete.
 * *public abstract class AbstractIterator<T> : Iterator<T> {

    override fun listIterator(): ListIterator<E> = ListIteratorImpl(0)

    override fun listIterator(index: Int): ListIterator<E> = ListIteratorImpl(index)

    /**
     * Fills the provided [array] or creates new array of the same type
     * and fills it with the elements of this collection.
     * * protected open fun <T> toArray(array: Array<T>): Array<T> = copyToArrayImpl(this, array)
     */

    Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*

package kotlin.collections

private open inner class ListIteratorImpl(index: Int) :

override fun hasNext(): Boolean = index < size

override fun next(): E {
    if

override fun get(index: Int): E {

}
this@AbstractMap.size
    }\n    return _keys!!\n}\n@kotlin.jvm.Volatile\nprivate var _keys: Set<K>? = null\n\noverride fun toString(): String = entries.joinToString("",", ", "[", "]") { toString(it) }\nprivate fun toString(entry: Map.Entry<K, V>): String = toString(entry.key) + "+=" + toString(entry.value)\n\noverride fun toString(o: Any?): String = if (o == null) "null" else o.toString()\n\n/**
 * Returns a read-only [Collection] of all values in this map.\n */\n*\n* Accessing this property first time creates a values view from [entries].\n* All subsequent accesses just return the created instance.\n* An override val values: Collection<V>\n    get() {
        if (_values == null) {
            _values = object : AbstractCollection<V>() {\n
                override operator fun contains(element: @UnsafeVariance V): Boolean = containsValue(element)\n                override operator fun iterator(): Iterator<V> = entryIterator()\n                override fun hasNext(): Boolean = entryIterator.hasNext()\n                override fun next(): V = entryIterator.next().value\n            }\n        }\n        return _values!!\n    }\n\nprivate fun implFindEntry(key: K): Map.Entry<K, V>? = entries.firstOrNull { it.key == key }\n\ninternal companion object {\n
    internal fun entryHashCode(e: Map.Entry<*, *>): Int = with(e) { (key?.hashCode() ?: 0) xor (value?.hashCode() ?: 0) }\n
    internal fun entryToString(e: Map.Entry<*, *>): String = with(e) { e.key + "=" + e.value }\n
    internal fun entryEquals(e: Map.Entry<*, *>, other: Any?): Boolean {\n        if (other !is Map.Entry<*, *>) return false\n        return e.key == other.key && e.value == other.value\n    }\n}\n
/**
 * Provides a skeletal implementation of the read-only [Set] interface.\n * This class is intended to help implementing read-only sets so it doesn't support concurrent modification tracking.\n */\n@kotlin.jvm.Volatile\nprivate var _values: Collection<V>? = null\n\nprivate fun implFindEntry(key: K): Map.Entry<K, V>? = null\n\n/**
 * Returns a single list of all elements from all arrays in the given array.\n */\n@kotlin.jvm.Vmln.package kotlin.collections\n\nimport kotlin.contracts.*
\n/**
 * Returns `true` if this nullable array is either null or empty.\n */\n@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\npublic inline fun Array<*>?.isNullOrEmpty(): Boolean {\n    contract {\n        returns(false) implies (this@isNullOrEmpty != null)\n    }\n
    return this == null || result.count { it.isNullOrEmpty() } == result.size\n}
can be found in the LICENSE.txt file.

NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).

NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).

NoSuchElementException when the map doesn't contain value for the property name and doesn't provide an implicit default (see [withDefault]).
MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =
    when (this) {
        is

MutableMapWithDefault -> this.map.withDefault(defaultValue)
        else -> MutableMapWithDefaultImpl(this, defaultValue)
    }

private interface MapWithDefault<K, out V> : Map<K, V>

private class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>):
    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean =
        map.containsValue(value)
        override fun getOrImplicitDefault(key: K): V = map.getOrDefault(key)
        override fun get(key: K): V? = map.get(key)
        override fun put(key: K, value: V): V? = map.put(key, value)
        override fun remove(key: K): V? = map.remove(key)
        override fun putAll(from: Map<out K, V>) = map.putAll(from)
        override fun clear() = map.clear()
        override fun getOrImplicitDefault(key: K) = map.getOrElse(key)
        val containsKey(key: K) = map.containsKey(key)
        override fun containsValue(value: @UnsafeVariance V) = map.containsValue(value)
        override fun getOrImplicitDefault(key: K): V? = map.getOrElseNullable(key)
        override fun getOrDefault(key: K): V = map.getOrElse(key, { default(key) })
        override fun getOrDefault(key: K, defaultValue: V): V = map.getOrElse(key, { defaultValue(key) })
        override fun getOrImplicitDefault(key: K): V? = map.getOrElse(key)
        override fun getOrDefault(key: K, defaultValue: V): V? = map.getOrElse(key, { defaultValue(key) })
    }

private class MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K, V>):
    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)
    override fun getOrImplicitDefault(key: K): V = map.getOrDefault(key)
    override fun get(key: K): V? = map.get(key)
    override fun put(key: K, value: V): V? = map.put(key, value)
    override fun remove(key: K): V? = map.remove(key)
    override fun putAll(from: Map<out K, V>) = map.putAll(from)
    override fun clear() = map.clear()
    override fun getOrImplicitDefault(key: K) = map.getOrElse(key)
    val containsKey(key: K) = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V) = map.containsValue(value)
    override fun getOrImplicitDefault(key: K): V? = map.getOrElseNullable(key)
    override fun getOrDefault(key: K, defaultValue: V): V? = map.getOrElse(key, { default(key) })
        override fun getOrImplicitDefault(key: K): V? = map.getOrElse(key)
        override fun getOrDefault(key: K, defaultValue: V): V? = map.getOrElse(key, { defaultValue(key) })
    }

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.Random

/**
 * Removes a single instance of the specified element from this collection, if it is present.
 * @return true if the element has been successfully removed; false if it was not present in the collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)

/**
 * Removes all of this collection's elements that are also contained in the specified collection.
 * @return true if any element was removed from the collection, false if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<T>.retainAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).retainAll(elements)

/**
 * Removes the element at the specified index from this list.
 * In Kotlin one should use the [MutableList.removeAt] function instead.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableList<T>.removeAt(index: Int): T =

Open Source Used In DNAC Platform Arica  12390
/*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.plusAssign(element: T) {
    this.add(element)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Iterable<T>) {
    this.addAll(elements)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Array<T>) {
    this.addAll(elements)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Sequence<T>) {
    this.addAll(elements)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.minusAssign(element: T) {
    this.remove(element)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Iterable<T>) {
    this.removeAll(elements)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) {
    this.removeAll(elements)
}*\@kotlin.internal.InlineOnly\public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) {
    this.removeAll(elements)
}

/**
 * Adds all elements of the given [elements] collection to this mutable collection.
 */
public fun <T> MutableList<T>.addAll(elements: Collection<T>): Boolean = filterInPlace(elements.asList(), this, true)

/**
 * Retains only elements of this [MutableList] that match the given [predicate].
 *
 * @return `true` if any element was removed from this collection, or `false` when no elements were removed and collection was not modified.
 */
public fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, this, false)

private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    if (this !is RandomAccess)
        return (this as MutableList<T>).filterInPlace(predicate, predicateResultToRemove)

    var writeIndex: Int = 0
    for (readIndex in 0..lastIndex) {
        val element = this[readIndex]
        if (predicate(element) == predicateResultToRemove)
            continue

        if (writeIndex != readIndex)
            this[writeIndex] =
element

        writeIndex++
    }
    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo
            writeIndex)
            removeAt(removeIndex)

        return true
    } else {
        return false
    }
}

/**
 * Removes all elements from this [MutableCollection] that are also contained in the given [elements] collection.
 */
public fun <T> MutableList<T>.removeAll(elements: Iterable<T>): Boolean {
    return removeAll(elements.convertToSetForSetOperationWith(this))
}

/**
 * Removes all elements from this [MutableCollection] that are also contained in the given [elements] sequence.
 */
public fun <T> MutableList<T>.removeAll(elements: Sequence<T>): Boolean {
    val set = elements.toHashSet()
    return set.isNotEmpty() && removeAll(set)
}

/**
 * Removes all elements from this [MutableCollection] that are also contained in the given [elements] array.
 */
public fun <T> MutableList<T>.removeAll(elements: Array<out T>): Boolean {
    if (elements.isNotEmpty())
        return removeAll(elements.toHashSet())
    else
        return retainNothing()
}

/**
 * Randomly shuffles elements in this mutable list using the specified [random] instance as the source of randomness.
 */
private fun MutableList<T>.retainNothing(): Boolean {
    val result = isNotEmpty()
    clear()
    return result
}

/**
 * Returns a new list with the elements of this list randomly shuffled using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.3")
public fun <T> Iterable<T>.shuffled(random: Random): List<T> = toMutableList().apply { shuffle(random) }

...
MutableList<T> = ReversedList(this)

"/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")
@file:UseExperimental(
ExperimentalTypeInference::class)

package kotlin.sequences

import kotlin.coroutines.*
import kotlin.coroutines.intrinsics.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Builds a [Sequence] lazily yielding values one by one.
 */
@SinceKotlin("1.3")
public fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
iterator(block)
}

/**
 * Builds an [Iterator] lazily yielding values one by one.
 */
@SinceKotlin("1.3")
@Deprecated("Use 'sequence { }' function instead.",
ReplaceWith("sequence(builderAction)"), level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
iterator(builderAction)
}

/**
 * The scope for yielding values of a [Sequence] or an [Iterator],
 * provides [yield] and [yieldAll] suspension functions.
 */
@RestrictsSuspension
@SinceKotlin("1.3")
public abstract class SequenceScope<in T> internal constructor() {
    /**
     * Yields a value to the [Iterator] being built and suspends
     * until the next value is requested.
     */
    public abstract suspend fun yield(value: T)

    /**
     * Yields all values from the `iterator` to the [Iterator] being built
     * and suspends until all these values are iterated and the next one is requested.
     */
    public abstract suspend fun yieldAll(iterator: Iterator<T>)

    /**
     * Yields potentially infinite sequence of values to the [Iterator]
     * being built and suspends until all these values are iterated and the next one is requested.
     */
    public suspend fun yieldAll(sequence: Sequence<T>) = yieldAll(sequence.iterator())
}

@Deprecated("Use SequenceScope class instead.",
ReplaceWith("SequenceScope<T>"), level = DeprecationLevel.ERROR)
public typealias SequenceBuilder<T> = SequenceScope<T>

private typealias State = Int

private const val State_NotReady: State = 0
private const val State_ManyNotReady: State = 1
private const val State_ManyReady: State = 2
private const val State_Done: State = 3
private const val State_Failed: State = 4

private class

Open Source Used In DNAC Platform Arica

* Buffer overflow is not allowed so [add] doesn't overwrite tail but raises an exception.

```kotlin
    override val size: Int get() = _size
}

/**
 * Provides ring buffer

    }

    override fun get(index: Int): E {
        checkElementIndex(index, _size)

        return
```

```
internal class MovingSubList<out E>(private val list: List<E>) : AbstractList<E>(), RandomAccess {

    fun move(fromIndex: Int, toIndex: Int) {

        buffer.removeFirst(step)
                }
                if (buffer.isNotEmpty()) yield(buffer)
            }
        }
        while (buffer.size > step) {
                    yield(if (reuseBuffer) buffer else ArrayList(buffer))
                    buffer.removeFirst(step)
                }
            }
            if (partialWindows) {

                    yield(if (reuseBuffer) buffer else 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
                    if (buffer.size < size) { buffer = 

                    buffer.add(e)
                if (buffer.isFull()) {
```

```
```
```
```kotlin
private class RingBuffer<T>(private val buffer: Array<*>?, filledSize: Int) : AbstractList<T>(), RandomAccess {
    private val defaultCapacity = capacity
    private val defaultCapacity = capacity

    private val defaultCapacity = capacity
    private val defaultCapacity = capacity

    override fun get(index: Int): T {
        checkElementIndex(index, size)
        return buffer[index] as T
    }

    override fun set(index: Int, element: T) {
        checkElementIndex(index, size)
        buffer[index] = element
    }

    override fun removeAt(index: Int) {
        checkElementIndex(index, size)
        val element = buffer[index]
        buffer[index] = buffer[--index] as T
        size--
    }

    override fun add(element: T) {
        if (isFull()) {
            throw IllegalStateException("ring buffer is full")
        } else {
            buffer[++index] = element
            size++
        }
    }

    override fun iterator(): Iterator<T> = object : AbstractIterator<T>() {
        private var index = startIndex

        override fun computeNext() {
            if (count == 0) {
                index = indexBit + 1
                count--
            } else {
                index = indexBit + 1
                count--
            }
        }
    }

    fun isFull() = size == capacity

    override fun emptyIterator(): AbstractIterator<T> = object : AbstractIterator<T>() {
        override fun computeNext() {
            if (count == 0) {
                return done()
            } else {
                setNext(buffer[index] as T)
                index = index + 1
                count--
            }
        }
    }

    fun expand(n: Int): RingBuffer<T> {
        val newCapacity = (capacity + (capacity shr 1)) + 1).coerceAtMost(maxCapacity)
        val newBuffer = if (startIndex == 0)
            buffer.copyOf(newCapacity) else ArrayNullsOf(newCapacity)
        return RingBuffer(newBuffer, size)
    }

    fun expanded(maxCapacity: Int): RingBuffer<T> {
        val newCapacity = (capacity + (capacity shr 1)) + 1).coerceAtMost(maxCapacity)
        val newBuffer = if (startIndex == 0)
            buffer.copyOf(newCapacity) else ArrayNullsOf(newCapacity)
        return RingBuffer(newBuffer, size)
    }

    fun resize(n: Int) {
        require(n >= 0) { "n shouldn't be negative but it is $n" }
        require(n <= size) {
            val size = n
            giveForward(n)
            size = n
        }
    }

    fun giveForward(n: Int) {
        require(n >= 0) { "n shouldn't be negative but it is $n" }
        require(n <= size) { "n shouldn't be greater than the buffer size: n = $n, size = $size" }
        if (n > 0) {
            val start = startIndex
            val end = start.forward(n)
            if (start < end) {
                buffer.fill(null, start, end)
                size = end - startIndex
                size = end - startIndex
            }
        }
    }

    fun isNull(): Boolean = size == 0

    fun nonNull(): Array<T> = buffer.copyOf(n) as Array<T>

    fun null(): Array<T> = buffer.copyOfNulls(n) as Array<T>

    fun nulls(): Array<T> = buffer.copyOfNulls(n) as Array<T>

    fun copyOf(n: Int): Array<T> = buffer.copyOf(n) as Array<T>

    fun copyOfNulls(n: Int): Array<T> = buffer.copyOfNulls(n) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyOfNulls(maxCapacity) as Array<T>

    fun copyOfNulls(size: Int): Array<T> = buffer.copyOfNulls(size) as Array<T>

    fun copyOfNulls(maxCapacity: Int): Array<T> = buffer.copyo
```kotlin
index - 1)\n    quickSort(array, left, index - 1)\n    if (index < right)\n        quickSort(array, index, right)\n\nUIntArray

@ExperimentalUnsignedTypes\nprivate fun partition(array: UIntArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot) i++
        while (array[j] > pivot) j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\nprivate fun quickSort(array: UIntArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

ULongArray

@ExperimentalUnsignedTypes\nprivate fun partition(array: ULongArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot) i++
        while (array[j] > pivot) j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\nprivate fun quickSort(array: ULongArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// Interfaces

/**
 * Sorts the given array using qsort algorithm.
 */
@ExperimentalUnsignedTypes\ninternal fun sortArray(array: UByteArray) = quickSort(array, 0, array.size - 1)
@ExperimentalUnsignedTypes\ninternal fun sortArray(array: UShortArray) = quickSort(array, 0, array.size - 1)
@ExperimentalUnsignedTypes\ninternal fun sortArray(array: UIntArray) = quickSort(array, 0, array.size - 1)
@ExperimentalUnsignedTypes\ninternal fun sortArray(array: ULongArray) = quickSort(array, 0, array.size - 1)

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.
 */

package kotlin.contracts

import kotlin.internal.ContractsDsl
import kotlin.internal.InlineOnly

/**
 * This marker distinguishes the experimental contract declaration API and is used to opt-in
 * for that feature when declaring contracts of user functions.
 * Any usage of a declaration annotated with `@ExperimentalContracts` must be accepted either
 * by `@UseExperimental` annotation, e.g.
 */
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
@Experimental
@MustBeDocumented
public annotation class ExperimentalContracts

/**
 * Provides a scope, where the functions of the contract DSL, such as `returns`,
 * `callsInPlace`, etc., can be used to describe the contract of a function.
 * This type is used as a receiver type of the lambda function passed to the `contract`
 * function.
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public interface ContractBuilder {
    /**
     * Describes a situation when a function returns normally, without any exceptions thrown.
     */
    @sample samples.contracts.returnsContract
    @ContractsDsl public fun returns(): Returns

    /**
     * Describes a situation when a function returns normally with the specified return value.
     */
    @sample samples.contracts.returnsTrueContract
    @sample samples.contracts.returnsFalseContract
    @sample samples.contracts.returnsNullContract
    @ContractsDsl fun returns(value: Any?): Returns

    /**
     * Describes a situation when a function returns normally with any value that is not null.
     */
    @sample samples.contracts.returnsNotNullContract
    @ContractsDsl fun returnsNotNull(): Returns

    /**
     * Describes a situation when a function returns normally with the specified return value.
     */
    @sample samples.contracts.returnsFalseContract
    @ContractsDsl public fun returnsNullContract()

    /**
     * Describes a situation when a function returns normally with any value that is not null.
     */
    @sample samples.contracts.returnsNotNullContract
    @ContractsDsl public fun returnsNotNull():
/**
 * Specifies that the function parameter \[\lambda\] is invoked in place.
 *
 * This contract specifies that
 * 1. the function \[\lambda\] can only be invoked during the call of the owner function,
 * and it won’t be invoked after that owner function call is completed;
 * 2. (optionally) the function \[\lambda\] is invoked the amount of times specified by the \[kind\] parameter,
 * see the \[InvocationKind\] enum for possible values.
 *
 * A function declaring the `callsInPlace` effect must be _inline_.
 */
@ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace
}

/**
 * Specifies how many times a function invokes its function parameter in place.
 *
 * See \[ContractBuilder.callsInPlace\] for the details of the call-in-place function contract.
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public enum class InvocationKind {
    /**
     * A function parameter will be invoked one time or not invoked at all.
     */
    AT_MOST_ONCE,

    /**
     * A function parameter will be invoked one or more times.
     */
    AT_LEAST_ONCE,

    /**
     * A function parameter will be invoked exactly one time.
     */
    EXACTLY_ONCE,

    /**
     * A function parameter is called in place, but it's unknown how many times it can be called.
     */
    UNKNOWN
}

/**
 * Specifies the contract of a function.
 *
 * The contract description must be at the beginning of a function and have at least one effect.
 * Only the top-level functions can have a contract for now.
 *
 * A function parameter will be invoked one time or not invoked at all.
 * A function parameter will be invoked exactly one time.
 * A function parameter will be invoked one or more times.
 * The contract description must be at the beginning of a function and have at least one effect.
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public interface ContinuationInterceptor : CoroutineContext.Element {
    /**
     * The key that defines *the* context interceptor.
     */
    companion object Key:

    /**
     * Returns continuation that wraps the original
     * continuation instance returned by \[interceptContinuation\] when the original
     * continuation completes and will not be used anymore. This function is invoked only if \[interceptContinuation\] had returned a different
     * continuation.
     */
    public fun <T>

    /**
     * Invoked for the continuation instance returned by \[interceptContinuation\] when the original
     * continuation completes and will not be used anymore. This function is invoked only if \[interceptContinuation\] had returned a different
     * continuation.
     */

continuation instance from the one it was invoked with.
*\n* Default implementation does nothing.
*\n* @param continuation Continuation instance returned by this interceptor's [interceptContinuation] invocation.
*\npublic fun releaseInterceptedContinuation(continuation: Continuation<*>) { /* do nothing by default
    // Performance optimization for a singleton Key
    @Suppress("UNCHECKED_CAST")
        if (key === Key) this as E else null
    // Performance optimization to a singleton Key
    public override fun minusKey(key: CoroutineContext.Key<*>) =
        if (key === Key) EmptyCoroutineContext else this
    // Performance optimization to a singleton Key
    public override fun plus(context: CoroutineContext): CoroutineContext =
        if (context === EmptyCoroutineContext) this else // fast path -- avoid lambda creation
            context.fold(this) { acc, element ->
                val removed = acc.minusKey(element.key)
                if (removed === EmptyCoroutineContext) element else {
                    // make sure interceptor is always last in the context (and thus is fast to get when present)
                    val interceptor = removed[ContinuationInterceptor]
                    if (interceptor == null) CombinedContext(removed, element) else {
                        val left = removed.minusKey(ContinuationInterceptor)
                        if (left === EmptyCoroutineContext)
                            CombinedContext(element, interceptor) else
                                CombinedContext(CombinedContext(left, element),
                                    interceptor)
                    }
                }
            }
    // Performance optimization to a singleton Key
    public override fun hashCode(): Int = 0
    public override fun toString(): String = "EmptyCoroutineContext"
}

//--------------------- internal impl ---------------------

// this class is not exposed, but is hidden inside implementations
// this is a left-biased list, so that `plus` works
naturally\n@SinceKotlin("1.3")\ninternal class CombinedContext\n    private val left: CoroutineContext,\n    private val element: Element\n    : CoroutineContext, Serializable \n    override fun <E : Element> get(key: Key<E>): E? \n    if (next is CombinedContext) \n        return if (next is CombinedContext) \n            next \n        else \n            return\n    operation(left.fold(initial, operation), element) \n    public override fun minusKey(key: Key<E>): CoroutineContext \n    get(key) \n    if (next is CombinedContext) \n        return next \n    else \n        return\n    private fun size(): Int \n        while (true) \n            cur = cur.left as? CombinedContext ?: return size \n        var size = 2 \n        while (true) \n            cur = cur.left as? CombinedContext ?: this \n            var size = 2 \n            while (true) \n                cur = cur.left as? CombinedContext ?: this \n                var size = 2 \n                while (true) \n                    cur = cur.left as? CombinedContext ?: this \n                    var size = 2 \n                    while (true) \n                        cur = cur.left as? CombinedContext ?: this \n                        var size = 2 \n                        while (true) \n                            cur = cur.left as? CombinedContext ?: this \n                            var size = 2
private fun contains(element: Element): Boolean = \n    get(element.key) == element
private fun containsAll(context: CombinedContext): Boolean = \n    contains(context.element)
override fun equals(other: Any?): Boolean = \n    this === other || other is CombinedContext && other.size() == size() &&
    other.containsAll(this)
override fun hashCode(): Int = left.hashCode() + element.hashCode()
override fun toString(): String = \n    "[" + fold("") { acc, element ->
        if (acc.isEmpty()) element.toString() else
            "$acc, $element"
    } + "]"
private fun writeReplace(): Any {\n    val n = size()
    val elements = arrayOfNulls<CoroutineContext>(n)
    var index = 0
    fold(Unit) { _, element -> elements
        if (acc.isEmpty()) element.toString() else
            "$acc, $element"
    }
    check(index == n)
    return
}
private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)

package kotlin.coroutines.intrinsics
import kotlin.coroutines.*
import kotlin.internal.InlineOnly
/**
 * Obtains the current continuation instance inside suspend functions and either suspends the execution
 * or returns the result immediately without suspension. If the [block] returns the special
 * COROUTINE_SUSPENDED value, it means that suspend function did suspend the execution and will not return
 * any result immediately. In this case, the [Continuation] provided to the [block] shall
 * be resumed by invoking [Continuation.resumeWith] at some moment in the future when the result becomes available to resume the computation.
 * The return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.
 * It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.
 * As the result type of the [block] is declared as `Any` and cannot be correctly type-checked, its proper return type
 * remains on the conscience of the suspend function's author. In this case, the [Continuation.intercepted] method
 * resumes coroutine directly in the invoker's thread without going through the [Continuation] provided to the [block] that might be present in the coroutine's [CoroutineContext].
 * It is the invoker's responsibility to ensure that a proper invocation context is established.
 * [Continuation.intercepted] can be used to acquire the intercepted continuation.
 * Note that it is not recommended to call either [Continuation.resume] or [Continuation.resumeWith] that might be present in the coroutine's [CoroutineContext].
 * @since 1.3
 * [suspendCoroutine] as a safer way to obtain current \n * instantiation instance.
 * @throws NotImplementedError([Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic] is not implemented)
 * @suspendCoroutineUninterceptedOrReturn \n * suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T =\n *    throw NotImplementedError("Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic")
 * This value is used as a return value of [suspendCoroutineUninterceptedOrReturn] `block` argument to state that the execution was not suspened and will not return any result immediately.
 */
in general code.** Using it outside of the context of\n * `suspendCoroutineUninterceptedOrReturn` function return value (including, but not limited to,\n * storing this value in other properties, returning it from other functions, etc.)\n* can lead to unspecified behavior of the code.\n */ It is implemented as property with getter to avoid
ProGuard <clinit> problem with multifeile IntrinsicsKt class

@SinceKotlin("1.3")
public val

COROUTINE_SUSPENDED: Any get() = CoroutineSingletons.COROUTINE_SUSPENDED

// Using enum here ensures two important properties:
//  1. It makes SafeContinuation serializable with all kinds of serialization
//    frameworks (since all of them natively support enums)
//  2. It improves debugging experience, since you clearly
//    see toString() value of those objects and what package they come from
@SinceKotlin("1.3")
@PublishedApi // This class is Published API via serialized representation of SafeContinuation, don't rename/move\ninternal enum
class CoroutineSingletons { COROUTINE_SUSPENDED, UNDECIDED, RESUMED }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package

/** Performs a bitwise AND operation between the two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.and(other: Byte): Byte =
    (this.toInt() and other.toInt()).toByte()\n
/** Performs a bitwise OR operation between the two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.or(other: Byte): Byte =
    (this.toInt() or other.toInt()).toByte()\n
/** Performs a bitwise XOR operation between the two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.xor(other: Byte): Byte =
    (this.toInt() xor other.toInt()).toByte()\n
/** Inverts the bits in this value.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Byte.inv(): Byte =
    (this.toInt().inv()).toByte()\n
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.experimental

/**
 * The experimental marker for type
 * inference augmenting annotations.
 *
 * Any usage of a declaration annotated with
 * `@ExperimentalTypeInference` must be accepted either by
 * annotating that usage with the [UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalTypeInference::class)` \n * or by using the compiler argument `-Xuse-experimental=kotlin.experimental.ExperimentalTypeInference`.
 */
@Experimental(level = Experimental.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.ANNOTATION_CLASS)
@SinceKotlin("1.3")
public annotation class

ExperimentalTypeInference

/**
 * Specifies that the constraint built for the type during type inference should be an equality one.
 */
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
internal annotation class

NoInfer\n
/**
 * Specifies that a corresponding member has the lowest priority in overload resolution.
 */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class

LowPriorityInOverloadResolution

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.internal

/**
 * Specifies that the corresponding type should be ignored during type inference.
 */
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
\n
\n
Open Source Used In DNAC Platform Arica  12400
overload resolution. Effectively this means that an extension annotated with this annotation will win in overload resolution over a member with the same signature.

```kotlin
@Retention(AnnotationRetention.BINARY)
internal annotation class HidesMembers

/**
 * The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).
 */
@Retention(AnnotationRetention.BINARY)
internal annotation class OnlyInputTypes

/**
 * Specifies that this declaration can have dynamic receiver type.
 */
@Retention(AnnotationRetention.BINARY)
internal annotation class DynamicExtension

/**
 * The value of this parameter should be a property reference expression (`this::foo`), referencing a `lateinit` property.
 * *(the backing field of which is accessible at the point where the corresponding argument is passed.*
 */
@Retention(AnnotationRetention.BINARY)
internal annotation class AccessibleLateinitPropertyLiteral

/**
 * Specifies which version should be compared with the [version] value, when compiling the usage of the annotated declaration.
 * Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported, so the Kotlin compiler before 1.2 is going to treat any [RequireKotlin] as if it requires the language version. Since 1.2, the Kotlin compiler supports
 * [RequireKotlinVersionKind.COMPILER_VERSION] and [RequireKotlinVersionKind.API_VERSION].
 * If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin),
 * Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a requirement
 * on the language version.
 * This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.
 */
@Retention(AnnotationRetention.SOURCE)
@Repeatable
@SinceKotlin("1.2")
internal annotation class RequireKotlin(
    val version: String,
    val message: String = "",
    val level: DeprecationLevel = DeprecationLevel.ERROR,
    val versionKind: RequireKotlinVersionKind = RequireKotlinVersionKind.LANGUAGE_VERSION,
    val errorCode: Int = -1
)

/**
 * The kind of the version that is required by [RequireKotlin].
 */
@SinceKotlin("1.2")
internal enum class RequireKotlinVersionKind {
    LANGUAGE_VERSION,
    COMPILER_VERSION,
    API_VERSION,
}
```

```kotlin
Delegates {
    /**
     * Returns a property delegate for a read/write property with a non-`null` value that is initialized not during
     * object construction time but at a later time.
     *
     * Returns a property delegate for a read/write property with a non-`null` value that is initialized not during
     * object construction time but at a later time.
     * @sample samples.properties.Delegates.notNullDelegate
     */
    public fun <T : Any> notNull():
```
ReadWriteProperty<Any?, T> = NotNullVar()

/**
 * Returns a property delegate for a read/write property
 * that calls a specified callback function when changed.
 * @param initialValue the initial value of the property.
 * @param onChange the callback which is called after the change of the property is made.
 * The value of the property
 * has already been changed when this callback is invoked.
 * @sample samples.properties.Delegates.observableDelegate
 * @sample samples.properties.Delegates.throwVetoableDelegate
 */
public inline fun <T> observable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):

    ReadWriteProperty<Any?, T> =
        object : ObservableProperty<T>(initialValue) {
            override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)
        }

/**
 * Returns a property delegate for a read/write property that calls a specified callback function when changed,
 * allowing the callback to veto the modification.
 * @param initialValue the initial value of the property.
 * @param onChange the callback which is called before a change to the property value is attempted.
 * The value of the property hasn't been changed yet, when this callback is invoked.
 * If the callback returns `true` the value of the property is being set to the new value,
 * and if the callback returns `false` the new value is discarded and the property remains its old value.
 * @sample samples.properties.Delegates.vetoableDelegate
 * @sample samples.properties.Delegates.throwVetoableDelegate
 */
public inline fun <T> vetoable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Boolean):

    ReadWriteProperty<Any?, T> =
        object : ObservableProperty<T>(initialValue) {
            override fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = onChange(property, oldValue, newValue)
        }

private class NotNullVar<T : Any>() : ReadWriteProperty<Any?, T> {
    private var value: T? = null

    public override fun getValue(thisRef: Any?, property: KProperty<*>): T {
        return value
    }

    public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {
        this.value = value
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Implements the core logic of a property delegate for a read/write property that calls callback functions when changed.
 * @param initialValue the initial value of the property.
 */
public abstract class ObservableProperty<T>(initialValue: T) :
    ReadWriteProperty<Any?, T> {

    /**
     * The callback which is called before a change to the property value is attempted.
     * The value of the property hasn't been changed yet, when this callback is invoked.
     * If the callback returns `true` the value of the property is being set to the new value,
     * and if the callback returns `false` the new value is discarded and the property remains its old value.
     * @param initialValue the initial value of the property.
     * @param protected open fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = true
     * @param protected open fun afterChange(property: KProperty<*>, oldValue: T, newValue: T): Unit
     */

    /**
     * The callback which is called after the change of the property is made.
     * The value of the property
     * has already been changed when this callback is invoked.
     * protected open fun afterChange(property: KProperty<*>, oldValue: T, newValue: T): Unit {
     * val return value
     */

    companion object [Random.Default]

    /*
    * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
    * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
    */

    package kotlin.random

    import kotlin.math.nextDown

    /**
     * An abstract class that is implemented by random number generator algorithms.
     */

    companion object [Random.Default]

    /*
     * Generates an `Int` whose lower [bitCount] bits are filled with random values and the remaining upper bits are zero.
     */

    companion object [Random.Default]

    /*
     * Gets the next random [bitCount] number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.
     */

    companion object [Random.Default]
random 'Int' from the random number generator.

* Generates a `Long` random value uniformly distributed between `Long.MIN_VALUE` and `Long.MAX_VALUE` (inclusive).
* Generates an 'Int' random value uniformly distributed between '0' (inclusive) and the specified [until] bound.
* Generates an 'Int' random value uniformly distributed between '0' (inclusive) and the specified [until] bound (exclusive).
* @param until must be positive.
* @throws IllegalArgumentException if [until] is negative or zero.
* public open fun nextLong(from: Int, until: Int): Long {
    checkRangeBounds(from, until)
    val n = until - from
    if (n > 0 || n == Int.MAX_VALUE) {
        val rnd = fastLog2(n) + nextBits(bitCount) / n
        return from + rnd
    } else {
        while (true) {
            val rnd = nextInt()
            if (rnd in from until until) return rnd
        }
    }
}

* Generates a 'Long' random value uniformly distributed between '0' (inclusive) and the specified [until] bound.
* Generates a 'Long' random value uniformly distributed between '0' (inclusive) and the specified [until] bound (exclusive).
* @param until must be positive.
* @throws IllegalArgumentException if [until] is greater than or equal to [until].
* public open fun nextLong(from: Int, until: Int): Long = nextLong(0, until)

Long
        = nextLong(0, until)

val n = until - from
if (n > 0 || n == Int.MAX_VALUE) {
    val rnd = if (n and -n == n) {
        val nLow = n.toInt()
        val nHigh = (n ushr 32).toInt()
        rnd = when {
            nLow != 0 -> {
                val bitCount = fastLog2(nLow)
                nextBits(bitCount).toLong() and 0xFFFF_FFFF
            }
            nLow == 1 ->
                        // toUInt().toLong()
                        nextInt().toLong() and 0xFFFF_FFFF
            } else {
                var v: Int
                do {
                    val bits = nextInt().ushr(1)
                        v = bits % n
                } while (bits - v + (n - 1) < 0)
                v
            }
        }
        return from + rnd
    } else {
        while (true) {
            val rnd = nextInt()
            if (rnd in from until until) return rnd
        }
    }
}

// toUint().toLong() // toUint().toLong() and 0xFFFF_FFFF
}

nextBits(bitCount).toLong().shl(32) + nextInt()

nHigh == 1 ->
                        // toUInt().toLong()
                        nextInt().toLong() and 0xFFFF_FFFF
else -> {
    val bitCount = fastLog2(nHigh)
    nextBits(bitCount).toLong().shl(32) + nextInt()
}

* Generates a 'Double' random value uniformly distributed between 0 (inclusive) and 1 (exclusive).
* Generates a 'Double' random value uniformly distributed between 0 (inclusive) and 1 (exclusive).
* @throws IllegalArgumentException if [until] is greater than or equal to [until].
* public open fun nextDouble(): Double = doubleFromParts(nextBits(26), nextBits(27))

Long
        = nextLong(0, until)

val n = until - from
if (n > 0 || n == Int.MAX_VALUE) {
    val rnd = if (n and -n == n) {
        val nLow = n.toInt()
        val nHigh = (n ushr 32).toInt()
        rnd = when {
            nLow != 0 -> {
                val bitCount = fastLog2(nLow)
                nextBits(bitCount)
            }
            nLow == 1 ->
                        // toUInt().toLong()
                        nextInt().toLong() and 0xFFFF_FFFF
            } else {
                var v: Int
                do {
                    val bits = nextInt().ushr(1)
                        v = bits % n
                } while (bits - v + (n - 1) < 0)
                v
            }
        }
        return from + rnd
    } else {
        while (true) {
            val rnd = nextInt()
            if (rnd in from until until) return rnd
        }
    }
}

nextDouble(0, until)

* Generates a 'Double' random value uniformly distributed between 0 (inclusive) and [until].
* Generates a 'Double' random value uniformly distributed between 0 (inclusive) and [until].
* @throws IllegalArgumentException if [from] is greater than or equal to [until].
* public open fun nextDouble(): Double = doubleFromParts(nextBits(26), nextBits(27))

val n = until - from
val r = if (size.isInfinite()) && from.isFinite() && until.isFinite()) {
    val r1 = nextDouble() * (until / 2 - from / 2)
        from + r1 + r1
    } else {

from + nextDouble() * size
    }
    return if (r >= until) until.nextDown() else r
}

    /**
     * Gets the next random [Float] value uniformly distributed between 0 (inclusive) and 1 (exclusive).
     */
    public open fun nextFloat(): Float = nextBits(24) / (1 shl 24).toFloat()

    /**
     * Fills a subrange of the specified byte [array] starting from [fromIndex] inclusive and ending [toIndex] exclusive with random bytes.
     */
    public open fun nextBytes(array: ByteArray, fromIndex: Int = 0, toIndex: Int = array.size): ByteArray
        return require(fromIndex in 0..array.size && toIndex in 0..array.size)

        val steps = (toIndex - fromIndex) / 4
        var position = fromIndex
        repeat(steps) {
            val v = nextInt()
            array[position] = v.toByte()
            array[position + 1] = v.ushr(8).toByte()
            array[position + 2] = v.ushr(16).toByte()
            array[position + 3] = v.ushr(24).toByte()
            position += 4
        }
        val remainder = toIndex - position
        val vr = nextBits(remainder * 8)
        for (i in 0 until remainder) {
            array[position + i] = vr.ushr(i * 8).toByte()
        }
        return array
    }

    /**
     * Fills the specified byte [array] with random bytes and returns it.
     */
    public open fun nextBytes(array: ByteArray): ByteArray = nextBytes(array, 0, array.size)

    /**
     * Creates a byte array of the specified [size], filled with random bytes.
     */
    public open fun nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))

    /**
     * The default random number generator.
     *
     * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.
     */
    companion object Default : Random() {
        private val defaultRandom: Random = defaultPlatformRandom()
        override fun nextBits(bitCount: Int): Int = defaultRandom.nextBits(bitCount)
        override fun nextInt(): Int = defaultRandom.nextInt()
        override fun nextInt(until: Int): Int = defaultRandom.nextInt(until)
        override fun nextInt(from: Int, until: Int): Int = defaultRandom.nextInt(from, until)
        override fun nextLong(): Long = defaultRandom.nextLong()
        override fun nextLong(until: Long): Long = defaultRandom.nextLong(until)
        override fun nextLong(from: Long, until: Long): Long = defaultRandom.nextLong(from, until)
        override fun nextBoolean(): Boolean = defaultRandom.nextBoolean()
        override fun nextDouble(): Double = defaultRandom.nextDouble()
        override fun nextDouble(until: Double): Double = defaultRandom.nextDouble(until)
        override fun nextDouble(from: Double, until: Double): Double = defaultRandom.nextDouble(from, until)
        override fun nextFloat(): Float = defaultRandom.nextFloat()
        override fun nextBytes(array: ByteArray): ByteArray = defaultRandom.nextBytes(array)
        override fun nextBytes(size: Int): ByteArray = defaultRandom.nextBytes(size)
        override fun nextBytes(array: ByteArray, fromIndex: Int, toIndex: Int): ByteArray = defaultRandom.nextBytes(array, fromIndex, toIndex)
        @Deprecated("Use Default companion object instead", level = DeprecationLevel.HIDDEN)
        @Suppress("DEPRECATION_ERROR")
        @kotlin.jvm.JvmField
    }

    public object Companion : Random() {
        override fun nextBits(bitCount: Int): Int = Default.nextBits(bitCount)
    }

    /**
     * Returns a repeatable random number generator seeded with the given [seed] `Int` value.
     */
    public fun Random(seed: Int): Random = XorWowRandom(seed, seed.shr(31))

    /**
     * Returns a repeatable random number generator seeded with the given [seed] `Long` value.
     */
    public fun Random(seed: Long): Random = XorWowRandom(seed.toInt(), seed.shr(32).toInt())

    /**
     * Generates an `Int` random value
uniformly distributed in the specified [range]:

```kotlin
@throws IllegalArgumentException if [range] is empty.

 */
@SinceKotlin("1.3")
public fun
Random.nextInt(range: IntRange): Int = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)
    else -> nextInt(range.first - 1, range.last) + 1
}

/**
 * Gets the next random 'Long' from the random number generator in the specified [range].
 * Generates a 'Long' random value uniformly distributed in the specified [range]:
 * from 'range.start' inclusive to 'range.endInclusive' inclusive.
 */
@SinceKotlin("1.3")
public fun
Random.nextLong(range: LongRange): Long = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1)
    range.first > Long.MIN_VALUE -> nextLong(range.first - 1, range.last) + 1
    else -> nextLong(range.first - 1, range.last) + 1
}

internal expect fun defaultPlatformRandom(): Random
internal expect fun
doubleFromParts(hi26: Int, low27: Int): Double

@UseExperimental(ExperimentalStdlibApi::class)
internal fun
fastLog2(value: Int): Int = 31 - value.countLeadingZeroBits()

/** Takes upper [bitCount] bits (0..32) from this number. */
internal fun Int.takeUpperBits(bitCount: Int): Int =
    this.ushr(32 - bitCount) and (-(bitCount).shr(31))

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    "Range is empty: ($from, $until)"
}

internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) {
    "Range is empty: ($from, $until)"
}

internal fun checkRangeBounds(from: Double, until: Double) =
    require(until > from) {
        "Range is empty: ($from, $until)"
    }

internal fun boundsErrorMessage(from: Any, until: Any) = "Range is empty: ($from, $until)"

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt =
    nextInt().toUInt()

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(until: UInt): UInt =
    nextUInt(0u, until)

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(from: UInt, until: UInt):
    UInt {
        val signedFrom = from.toInt() xor Int.MIN_VALUE
        val
```
IllegalArgumentException if \[\text{until}\] is zero.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public

fun Random.nextULong(until: ULong): ULong = nextULong(0uL, until)

/**
 * Gets the next random \[ULong\] value uniformly distributed in the specified \[range]\.
 * @throws
 * IllegalArgumentException if \[from\] is greater than or equal to \[until]\.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(from: ULong, until: ULong): ULong {
    checkULongRangeBounds(from, until)

    val signedFrom = from.toLong() xor Long.MIN_VALUE
    val signedUntil = until.toLong() xor Long.MIN_VALUE

    val signedResult = nextLong(signedFrom, signedUntil) xor Long.MIN_VALUE
    return signedResult.toULong()
}

/**
 * Creates an unsigned byte \[array\] of the specified \[size\], filled with random bytes.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUBytes(size: Int): UByteArray

/**
 * Fills a subrange of the specified \[UByte\] \[array\] starting from \[fromIndex\] inclusive and ending \[toIndex\] exclusive with random \[UBytes\].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUBytes(array: UByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray

package kotlin.random

/**
 * Random number generator, algorithm "xorwow" from p. 5 of Marsaglia, "Xorshift RNGs".
 * Cycles after \(2^{160} \times (2^{32}-1)\) repetitions.
 */
internal class XorWowRandom
internal constructor(

private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private

var addend: Int
) : Random() {

    internal constructor(seed1: Int, seed2: Int) :
            this(seed1, seed2, 0, 0,

    init {
        require((x or y or z or w or v) != 0) { "Initial state must have at least one non-zero element." }

        // some trivial seeds can produce several values with zeroes in upper bits, so we discard first 64
        repeat(64) { nextInt() }
    }

    override fun nextInt(): Int {
        var t =
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        return t + addend
    }

    override fun nextBits(bitCount: Int): Int =
        nextInt().takeUpperBits(bitCount)
IEEE-754 comparison order instead of total order of floating point numbers.

```kotlin
interface ClosedFloatingPointRange<T : Comparable<T>> : ClosedRange<T> {
    override fun contains(value: T): Boolean = lessThanOrEquals(start, value) && lessThanOrEquals(value, endInclusive)
    override fun isEmpty(): Boolean = !lessThanOrEquals(start, endInclusive)

    /**
     * Compares two values of range domain type and returns true if first is less than or equal to second.
     */
    fun lessThanOrEquals(a: T, b: T): Boolean {
        return a in start..endInclusive
    }

    /**
     * Represents a range of [Comparable] values.
     * @param start the start of the range.
     * @param endInclusive the end of the range (inclusive).
     */
    private class ComparableRange<T : Comparable<T>>(
        override val start: T,
        override val endInclusive: T
    ) : ClosedRange<T> {
        override fun equals(other: Any?): Boolean {
            return other is ComparableRange<*> && (isEmpty() && other.isEmpty() ||
                    start == other.start && endInclusive == other.endInclusive)
        }

        override fun hashCode(): Int {
            return if (isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()
        }

        override fun toString(): String = "$start..$endInclusive"
    }

    /**
     * Creates a range from this [Comparable] value to the specified [that] value.
     * This value needs to be smaller than [that] value, otherwise the returned range will be empty.
     * @param that the end value of the range.
     */
    public operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)

    /**
     * Creates a range from this [Double] value to the specified [that] value.
     * Numbers are compared with the ends of this range according to IEEE-754.
     * @param that the end value of the range.
     */
    @SinceKotlin("1.1")
    public operator fun Double.rangeTo(that: Double): ClosedFloatingPointRange<Double> = ClosedDoubleRange(start, endInclusive)
}
```

Returns a runtime representation of the given reified type [T] as an instance of [KType].
// Open Source Used In DNAC Platform Arica 12409

package kotlin.text

/**
 * Represents a collection of captured groups in a single match of a regular
 * expression.
 *
 * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the
 * regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the
 * entire match.
 *
 * An element of the collection at the particular index can be `null` if the corresponding
 * group in the regular expression is optional and there was no match captured by that group.
 * There was no match captured by that group.
 * There was no match captured by that group.
 */
public interface MatchGroupCollection : Collection<MatchGroup?> {

    /** Returns a group with the specified [index].
     *
     * @return An instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.
     */
    public operator fun get(index: Int): MatchGroup?
}

/**
 * Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.
 */
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
    /**
     * Returns a named group with the specified [name].
     * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.
     */
    public operator fun get(name: String): MatchGroup?
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /** The range of indices in the original string where match was captured. */
    public val range: IntRange
    /** The substring from the input string captured by this match. */
    public val value: String
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The character & */
    public const val amp: Char = '\u0026'
    /** The character < */
    public const val less: Char = '\u003C'
    /** The character > */
    public const val greater: Char = '\u003E'
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The character × */
    public const val times: Char = '\u00D7'
    /** The character ¢ */
    public const val cent: Char = '\u00A2'
    /** The character £ */
    public const val pound: Char = '\u00A3'
    /** The character § */
    public const val section: Char = '\u00A7'
    /** The character © */
    public const val copyright: Char = '\u00A9'
    /** The character « */
    public const val leftGuillemete: Char = '\u00AB'
    /** The character » */
    public const val rightGuillemete: Char = '\u00BB'
    /** The character ® */
    public const val registered: Char = '\u00AE'
    /** The character ° */
    public const val degree: Char = '\u00B0'
    /** The character ± */
    public const val plusMinus: Char = '\u00B1'
    /** The character ¶ */
    public const val paragraph: Char = '\u00B6'
    /** The character · */
    public const val middleDot: Char = '\u00B7'
    /** The character ½ */
    public const val half: Char = '\u00BD'
    /** The character – */
    public const val ndash: Char = '\u2013'
    /** The character — */
    public const val mdash: Char = '\u2014'
    /** The character ' */
    public const val leftSingleQuote: Char = '\u2018'
    /** The character ' */
    public const val rightSingleQuote: Char = '\u2019'
    /** The character ' */
    public const val lowSingleQuote: Char = '\u201A'
    /** The character " */
    public const val leftDoubleQuote: Char = '\u201C'
    /** The character " */
    public const val rightDoubleQuote: Char = '\u201D'
    /** The character " */
    public const val lowDoubleQuote: Char = '\u201E'
    /** The character † */
    public const val dagger: Char = '\u2020'
    /** The character ‡ */
    public const val doubleDagger: Char = '\u2021'
    /** The character • */
    public const val bullet: Char = '\u2022'
    /** The character … */
    public const val ellipsis: Char = '\u2026'
    /** The character ′ */
    public const val prime: Char = '\u2032'
    /** The character ″ */
    public const val doublePrime: Char = '\u2033'
    /** The character € */
    public const val euro: Char = '\u20AC'
    /** The character ™ */
    public const val tm: Char = '\u2122'
    /** The character ≈ */
    public const val almostEqual: Char = '\u2248'
    /** The character ≠ */
    public const val notEqual: Char = '\u2260'
    /** The character ≤ */
    public const val lessOrEqual: Char = '\u2264'
    /** The character ≥ */
    public const val greaterOrEqual: Char = '\u2265'
}
/*
 * Copyright 2010-2019
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

/**
 * A collection of groups matched by the regular expression.
 * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
 */
public val groupCollection: MatchGroupCollection

/**
 * A list of matched indexed group values.
 * This list has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
 * If the group in the regular expression is optional and there were no match captured by that group,
 * corresponding item in [groupValues] is an empty string.
 */
public val groupValues: List<String>

/**
 * An instance of [MatchResult.Destructured] wrapper providing components for destructuring assignment of group values.
 * Component1 corresponds to the value of the first group, component2 corresponds to the value of the second, and so on.
 * If the group in the regular expression is optional and there were no match captured by that group,
 * corresponding component value is an empty string.
 */
public val destructured: Destructured get() = Destructured(this)

/** Returns a new [MatchResult] with the results for the next match, starting at the position at which the last match ended (at the character after the last matched character).
 * @sample samples.text.Regexps.matchDestructuringToGroupValues
 */
public fun next(): MatchResult?

/**
 * Provides components for destructuring assignment of group values.
 * Component1 corresponds to the value of the first group, and so on.
 * If the group in the regular expression is optional and there were no match captured by that group,
 * corresponding component value is an empty string.
 * @sample samples.text.Regexps.matchDestructuringToGroupValues
 */
public class Destructured internal constructor(public val match: MatchResult) {
    @kotlin.internal.InlineOnly
    public operator inline fun component1(): String = match.groupValues[1]

    @kotlin.internal.InlineOnly
    public operator inline fun component2(): String = match.groupValues[2]

    @kotlin.internal.InlineOnly
    public operator inline fun component3(): String = match.groupValues[3]

    @kotlin.internal.InlineOnly
    public operator inline fun component4(): String = match.groupValues[4]

    @kotlin.internal.InlineOnly
    public operator inline fun component5(): String = match.groupValues[5]

    @kotlin.internal.InlineOnly
    public operator inline fun component6(): String = match.groupValues[6]

    @kotlin.internal.InlineOnly
    public operator inline fun component7(): String = match.groupValues[7]

    @kotlin.internal.InlineOnly
    public operator inline fun component8(): String = match.groupValues[8]

    @kotlin.internal.InlineOnly
    public operator inline fun component9(): String = match.groupValues[9]

    @kotlin.internal.InlineOnly
    public operator inline fun component10(): String = match.groupValues[10]

    /** Returns destructured group values as a list of strings.
     * @sample samples.text.Regexps.matchDestructuringToGroupValues
     */
    public fun toList(): List<String> = match.groupValues.subList(1, match.groupValues.size)
}

/**
 * A source of time for measuring time intervals.
 * The only operation provided by the clock is [markNow]. It returns a [ClockMark], which can be used to query the elapsed time later.
 * The operation provided by the clock is [markNow]. It returns a [ClockMark], which can be used to query the elapsed time later.
 * @see [measureTime]
 * @see [measureTimedValue]
 */
@SinceKotlin("1.3")
@ExperimentalTime
public interface Clock {
    /**
     * Marks a time point on this clock.
     * The returned [ClockMark] instance encapsulates captured time point and allows querying the duration of time interval [elapsed] from that point.
     * @see [measureTime]
     * @see [measureTimedValue]
     */
    public fun markNow(): ClockMark

    /**
     * Returns a clock mark on the same clock that is ahead of
     * this clock mark by the specified [duration].
     * The returned clock mark is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative.
     */
    public abstract fun elapsedNow(): Duration

    /**
     * Returns the amount of time passed from this clock mark on the clock from which this mark was taken.
     * Note that the value returned by this function can change on subsequent invocations.
     */
    public abstract fun markNow(): ClockMark

    /** Represents a time point notched on a particular [Clock].
     * Marks a time point on this clock.
     * The returned [ClockMark] instance encapsulates captured time point and allows querying the duration of time interval [elapsed] from that point.
     * @see [measureTime]
     * @see [measureTimedValue]
     */
    abstract class ClockMark {
        /**
         * Returns a clock mark on the same clock that is ahead of this clock mark by the specified [duration].
         * The returned clock mark is more _late_ when the [duration] is positive, and more _early_ when the [duration] is negative.
         */
        public open operator fun plus(duration: Duration): ClockMark
    }
}
/**
 * Returns a clock mark on the same clock that is behind this clock mark by the specified [duration].
 * The returned clock mark is more _early_, when the [duration] is positive, and more _late_ when the [duration] is negative.
 */
public open operator fun minus(duration: Duration): ClockMark = plus(-duration)

/**
 * Returns true if this clock mark has passed according to the clock from which this mark was taken.
 * Note that the value returned by this function can change on subsequent invocations.
 * If the clock is monotonic, it can change only from 'false' to 'true', namely, when the clock mark becomes behind the current point of the clock.
 * If the clock is non-monotonic, it can change from 'false' to 'true', namely, when the clock mark becomes behind the current point of the clock.
 */
public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()

/**
 * Returns false if this clock mark has not passed according to the clock from which this mark was taken.
 * Note that the value returned by this function can change on subsequent invocations.
 * If the clock is monotonic, it can change only from 'true' to 'false', namely, when the clock mark becomes behind the current point of the clock.
 */
public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()

@ExperimentalTime
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Deprecated("Subtracting one ClockMark from another is not a well defined operation because these clock marks could have been obtained from the different clocks.", level = DeprecationLevel.ERROR)
public inline operator fun ClockMark.minus(other: ClockMark): Duration = throw Error("Operation is disallowed.")

@ExperimentalTime
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Deprecated("Comparing one ClockMark to another is not a well defined operation because these clock marks could have been obtained from the different clocks.", level = DeprecationLevel.ERROR)
public inline operator fun ClockMark.compareTo(other: ClockMark): Int = throw Error("Operation is disallowed.")

private class AdjustedClockMark(val mark: ClockMark, val adjustment: Duration) : ClockMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): ClockMark = AdjustedClockMark(mark, adjustment + duration)
}

package kotlin.time

/**
 * The most precise clock available in the platform.
 *
 * The clock returns its readings from a source of monotonic time when it is available in a target platform, and resorts to a non-monotonic time source otherwise.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public expect object MonoClock : Clock

/**
 * An abstract class used to implement clocks that return their readings as [Long] values in the specified [unit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public abstract class AbstractLongClock(protected val unit: DurationUnit) : Clock {
    /**
     * This protected method should be overridden to return the current reading of the clock expressed as a [Long] number in the unit specified by the [unit] property.
     */
    protected abstract fun read(): Long

    private class LongClockMark(private val startedAt: Long, private val clock: AbstractLongClock, private val offset: Duration) : ClockMark() {
        override fun elapsedNow(): Duration = (clock.read() - startedAt).toDuration(clock.unit) - offset
        override fun plus(duration: Duration): ClockMark = LongClockMark(startedAt, clock, offset + duration)
    }

    override fun markNow(): ClockMark = LongClockMark(read(), this, Duration.ZERO)
}

/**
 * A clock that has programmatically updatable readings. It is useful as a predictable source of time in tests.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public abstract class AbstractDoubleClock(protected val unit: DurationUnit) : Clock {
    /**
     * This protected method should be overridden to return the current reading of the clock expressed as a [Double] number in the unit specified by the [unit] property.
     */
    protected abstract fun read(): Double

    private class DoubleClockMark(private val startedAt: Double, private val clock: AbstractDoubleClock, private val offset: Duration) : ClockMark() {
        override fun elapsedNow(): Duration = (clock.read() - startedAt).toDuration(clock.unit) - offset
        override fun plus(duration: Duration): ClockMark = DoubleClockMark(startedAt, clock, offset + duration)
    }

    override fun markNow(): ClockMark = DoubleClockMark(read(), this, Duration.ZERO)
}

@ExperimentalTime
private class AdjustedDoubleClockMark(val mark: ClockMark, val adjustment: Duration) : ClockMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): ClockMark = AdjustedDoubleClockMark(mark, adjustment + duration)
}

override fun ClockMark.plusAssign(duration: Duration): ClockMark = AdjustedDoubleClockMark(this, duration)

override fun ClockMark.minusAssign(duration: Duration): ClockMark = AdjustedDoubleClockMark(this, duration)
clock = TestClock()
* clock += 10.seconds
*/
*
* Advances the current reading value of this clock by the specified duration.
* When delta is greater than Long.MAX_VALUE or less than Long.MIN_VALUE, add it as double
val newReading = reading + (delta.toDouble()).toLong()
else { // when delta is greater than long, add it as double
val newReading = reading + delta
}

private fun overflow(duration: Duration) {
throw IllegalStateException("TestClock will overflow if its reading 
will be advanced by $duration.")
}

/*
public interface WallClock {
    fun currentTimeMilliseconds(): Long
}
*/

package kotlin.time

import kotlin.math.abs

@UseExperimental(ExperimentalTime::class)
private inline val storageUnit get() =
DurationUnit.NANOSECONDS

/**
 * Represents the amount of time one instant of time is away from another instant.
 * An infinite duration value [Duration.INFINITE] can be used to represent infinite timeouts.
 * To construct a duration use either the extension function [toDuration],
 * or the extension properties [hours], [minutes], [seconds],
 * and so on, available on [Int], [Long], and [Double] numeric types.
 * To get the value of this duration expressed in a particular [duration units]
 * use the functions [toInt], [toLong], and [toDouble]
 * or the properties [inHours], [inMinutes], [inSeconds], [inNanoseconds], and so on.
 * @constructor
 * @see Duration
 */
@SinceKotlin("1.3")
@ExperimentalTime
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_FOR_INLINE_CLASS")
public inline class Duration internal constructor(internal val value: Double) :
Comparable<Duration> {

//    init {
//        require(_value.isNaN().not())
//    }

    companion object {
        /** The duration equal to exactly 0 seconds. */
        public val ZERO: Duration = Duration(0.0)

        /** The duration whose value is positive infinity. It is useful
         * for representing timeouts that should never expire. */
        public val INFINITE: Duration =
Duration(Double.POSITIVE_INFINITY)

        /** Converts the given time duration [value] expressed in the specified [sourceUnit]
        * into the specified [targetUnit]. */
        public fun convert(value: Double, sourceUnit:
DurationUnit, targetUnit: DurationUnit):
Duration =
DurationUnit.convertDurationUnit(value, sourceUnit, targetUnit)

        // arithmetic operators
        /** Returns the negative of this value. */
        public operator fun unaryMinus():
Duration = Duration(-value)

        /** Returns a duration whose value is the sum of this and [other] duration values. */
        public operator fun plus(other: Duration):
Duration = Duration(value + other.value)

        /** Returns a duration whose value is this duration value multiplied by the given [scale] number. */
        public operator fun times(scale: Int):
Duration = Duration(value * scale)

        // accessor properties
        /** Returns the duration represented as a [Long] number of nanoseconds. */
        public val inNanoseconds: Long get() =
(sourceUnit.convertDurationUnit(storageUnit, unit).value).toLong()
@PublishedApi
    internal val nanosecondsComponent: Int get() = (inNanoseconds % 1e9).toInt() // conversion to units

    /** Returns the value of this duration expressed as a [Double] number of the specified [unit]. */
    fun toDouble(unit: DurationUnit): Double = convertDurationUnit(value, storageUnit, unit)

    // conversion to units

    /** Returns the value of this duration expressed as a [Long] number of the specified [unit]. */
    fun toLong(unit: DurationUnit): Long = toDouble(unit).toLong() // ** Returns the value of this duration expressed as an [Int] number of the specified [unit].

    /** The value of this duration expressed as a [Double] number of hours. */
    fun toDouble(): Double = toDouble(DurationUnit.HOURS)

    /** The value of this duration expressed as a [Double] number of minutes. */
    fun toDouble(): Double = toDouble(DurationUnit.MINUTES)

    /** The value of this duration expressed as a [Double] number of seconds. */
    fun toDouble(): Double = toDouble(DurationUnit.SECONDS)

    /** The value of this duration expressed as a [Long] number of milliseconds. */
    fun toLong(): Long = toDouble(DurationUnit.MILLISECONDS)

    /** The value of this duration expressed as a [Long] number of microseconds. */
    fun toLong(): Long = toDouble(DurationUnit.MICROSECONDS)

    /** The value of this duration expressed as a [Long] number of nanoseconds. */
    fun toLong(): Long = toDouble(DurationUnit.NANOSECONDS)

    // shortcuts

    /**
     * Returns the value of this duration expressed as a [Long] number of nanoseconds. * The value of this duration expressed as a [Long] number of milliseconds. * The value of this duration expressed as a [Long] number of microseconds. * The value of this duration expressed as a [Long] number of nanoseconds. * Returns the value of this duration expressed as a [Long] number of nanoseconds. * Returns the value of this duration expressed as a [Long] number of nanoseconds. * Returns the value of this duration expressed as a [Long] number of nanoseconds. * Returns the value of this duration expressed as a [Long] number of nanoseconds. */

    /**
     * Returns the string representation of this duration value expressed in the unit which yields the most compact and readable number value. * Special cases: * - zero duration is formatted as "\0s" * - the infinite duration is formatted as "\Infinity\" without unit * - very small durations (less than 1e-15 s) are expressed in seconds and formatted in scientific notation * - very big durations (more than 1e+7 days) are expressed in days and formatted in scientific notation * @return the value of duration in the automatically determined unit followed by that unit abbreviated name: d', h', m', s', ms', us', or ns. */
    override fun toString(): String = when {
        isInfinite() -> value.toString()
        value == 0.0 -> "0s"
        else -> {
            val absNs = absoluteValue.inNanoseconds
            var scientific = false
            var maxDecimals = 0
            val unit = when {
                absNs < 1e-6 -> DurationUnit.SECONDS
                absNs < 1e6 -> DurationUnit.MICROSECONDS
                absNs < 1000e9 -> DurationUnit.SECONDS
                absNs < 60_000e9 -> DurationUnit.MINUTES
                absNs < 3600_000e9 -> DurationUnit.HOURS
                absNs < 86400e9 -> DurationUnit.DAYS
                else -> DurationUnit.DAYS
            }
            val value = toDouble(unit)
            when (unit)
            val scientific -> formatScientific(value)
            val maxDecimals = 0 -> formatUptoDecimals(value, maxDecimals)
            else -> formatToExactDecimals(value, precision(abs(value)))
            var val unit = when ( byteArray.value < 1e9 -> DurationUnit.SECONDS
            else -> DurationUnit.MICROSECONDS
            var absNs < 1e19 -> DurationUnit.SECONDS
            else -> DurationUnit.MINUTES
            else -> DurationUnit.HOURS
            else -> DurationUnit.DAYS
            var scientific -> formatScientific(value)
            maxDecimals = 0 -> formatUptoDecimals(value, maxDecimals)
            else -> formatToExactDecimals(value, precision(abs(value)))
            var unit.shortName() unit = when (value < 1e9 -> 3
            value < 10 -> 2
            else -> 0
            )
            * Returns a string representation of this duration value expressed in the given [unit]. * and formatted with the specified [decimals] number of digits after
decimal point.
*\n* Special cases:
* - the infinite duration is formatted as `"Infinity"` without unit
*\n* @return the value of duration in the specified [unit] followed by that unit abbreviated name: `d`, `h`, `m`, `s`, `ms`, `us`, or `ns`.
*\n* @throws IllegalArgumentException if [decimals] is less than zero.
*\n* @sample samples.time.Durations.toStringDecimals
*\n* Returns an ISO-8601 based string representation of this duration.
*\n* The returned value is presented in the format `"PThHmMs.fS"`, where `h`, `m`, `s` are the integer components of this duration (see [toComponents]) and `f` is a fractional part of second. Depending on the roundness of the value the fractional part can be formatted with either `0`, `3`, or `9` decimal digits.
*\n* If the hours component absolute value of this duration is greater than [Int.MAX_VALUE], it is replaced with
[Int.MAX_VALUE].
*\n* so the infinite duration is formatted as `"PT2147483647H"`.
*\n* Negative durations are indicated with the sign `-` in the beginning of the returned string, for example, `"-PT5M30S"`.
*\n* @sample samples.time.Durations.toIsoString
*\n* Returns a `Duration` equal to this `Int` number of the specified [unit].
*\n* Returns a `Duration` equal to this `Long` number of the specified [unit].
*\n* Returns a `Duration` equal to this `Double` number of the specified [unit].
*\n* Returns a `Duration` equal to this `Int` number of nanoseconds.
*\n* Returns a `Duration` equal to this `Long` number of nanoseconds.
*\n* Returns a `Duration` equal to this `Double` number of nanoseconds.
*\n* Returns a `Duration` equal to this `Int` number of microseconds.
*\n* Returns a `Duration` equal to this `Long` number of microseconds.
*\n* Returns a `Duration` equal to this `Double` number of microseconds.
toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Int] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.seconds get() =
toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Long] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.seconds get() =
toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Double] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.seconds get() =
toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Int] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.minutes get() =
toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Long] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.minutes get() =
toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Double] number of minutes. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.minutes get() =
toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Int] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.hours get() =
toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Long] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.hours get() =
toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Double] number of hours. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.hours get() =
toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Int] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.days get() =
toDuration(DurationUnit.DAYS)

/** Returns a [Duration] equal to this [Long] number of days. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.days get() =
toDuration(DurationUnit.DAYS)

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
@kotlin.internal.InlineOnly
public inline operator fun Int.times(duration: Duration): Duration = duration * this

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
@SinceKotlin("1.3")
@ExperimentalTime
@kotlin.internal.InlineOnly
public inline operator fun Double.times(duration: Duration): Duration = duration * this

internal expect fun formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double, decimals: Int): String

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package kotlin.time

/**
 * The list of possible time measurement units, in which a duration can be expressed.
 *
 * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24
 * [HOURS].
 */
@SinceKotlin("1.3")
@ExperimentalTime
internal expect enum class DurationUnit {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS,
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS,
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILLISECONDS,
    /**
     * Time unit representing one second.
     */
    SECONDS,
    /**
     * Time unit representing one minute.
     */
    MINUTES,
    /**
     * Time unit representing one hour.
     */
    HOURS,
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS;
}

/** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified
 * [targetUnit]. */
@SinceKotlin("1.3")
@ExperimentalTime
internal fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double

internal fun DurationUnit.shortName(): String = when
Open Source Used In DNAC Platform Arica 12417

/**
 * This annotation marks the experimental preview of the standard library API for measuring time and working with durations.
 * Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library.
 * Any usage of a declaration annotated with `@ExperimentalTime` must be accepted either by annotating that usage with the [UseExperimental] annotation, e.g. `@UseExperimental(ExperimentalTime::class)`, or by using the compiler argument `-Xuse-experimental=kotlin.time.ExperimentalTime`.
*/
@Experimental(level = Experimental.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(
    CLASS,
    ANNOTATION_CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
)
@SinceKotlin("1.3")
public annotation class ExperimentalTime

/**
 * Executes the given function [block] and returns the duration of elapsed time interval.
 * The elapsed time is measured with [MonoClock].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return MonoClock.measureTime(block)
}

/**
 * Executes the given function [block] and returns the duration of elapsed time interval.
 * The elapsed time is measured with the specified `this` [Clock] instance.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun Clock.measureTime(block: () -> Unit): Duration {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    val mark = markNow()
    block()
    return mark.elapsedNow()
}

/**
 * Executing the given function [block] and returns an instance of [TimedValue] class, containing both the result of function execution and the duration of elapsed time interval.
 * The elapsed time is measured with the specified `this` [Clock] instance.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun <T> measureTimedValue(block: () -> T): TimedValue<T> {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return MonoClock.measureTimedValue(block)
}

/**
 * Returns a hash code value for the object or zero if the object is `null`.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun Any?.hashCode(): Int = this?.hashCode() ?: 0


components of a version, they must be non-negative and not greater than 255
(MAX_COMPONENT_VALUE).@n @constructor Creates a version from all three components.

@SinceKotlin("1.1") public class KotlinVersion(val major: Int, val minor: Int, val patch: Int) :
Comparable<KotlinVersion> { @n /*@n * Creates a version from [major] and [minor] components, leaving
[patch] component zero. @n */@n public constructor(major: Int, minor: Int) : this(major, minor, 0)
private fun versionOf(major: Int, minor: Int, patch: Int): Int {
require(major in 0..MAX_COMPONENT_VALUE && minor in 0..MAX_COMPONENT_VALUE && patch in
0..MAX_COMPONENT_VALUE) {
"Version components are out of range: $major.$minor.$patch"
} @n override fun toString(): String = "$major.$minor.$patch
override fun equals(other: Any?): Boolean { @n if (this == other) return true @n val otherVersion = (other as? KotlinVersion) ?: return false
return this.version == otherVersion.version
}@n override fun hashCode(): Int = version
override fun compareTo(other: KotlinVersion): Int = version - other.version
public fun isAtLeast(major: Int, minor: Int): Boolean = this.major > major || (this.major == major &&
(this.minor > minor || this.minor == minor &&
(this.patch >= patch))
public fun isAtLeast(major: Int, minor: Int, patch: Int): Boolean = this.major > major || (this.major == major &&
(this.minor >= minor &&
(this.patch >= patch))
public const val MAX_COMPONENT_VALUE = 255
public val CURRENT: KotlinVersion = KotlinVersion(1, 3, 61)
@file:kotlin.jvm.JvmName("LateinitKt")
@file:Suppress("unused")
package kotlin
import kotlin.internal.InlineOnly
import kotlin.internal.AccessibleLateinitPropertyLiteral
import kotlin.reflect.KProperty0

/**
 * Returns `true` if this lateinit property has been assigned a value, and `false`
 * otherwise.
 */
@SinceKotlin("1.2")
@InlineOnly
inline val @receiver:AccessibleLateinitPropertyLiteral
KProperty0<*>::isInitialized: Boolean
    get() = throw NotImplementedError("Implementation is
intrinsic")

/**
 * Represents a value with lazy initialization.
 *
 * To create an instance of
[Lazy] use the [lazy] function.
 */
public interface Lazy<out T> {

    /**
     * Gets the lazily initialized value of
the current Lazy instance.
     * Once the value was initialized it must not change during the rest of lifetime of this
Lazy instance.
     */
    public val value: T

    /**
     * Returns `true` if a value for this Lazy instance has been assigned a value, and `false`
otherwise.
 */
    public fun isInitialized(): Boolean

    /**
     * Specifies how a [Lazy] instance synchronizes initialization
among multiple threads.
 */
    @n @n @n // Note: Must be placed before CURRENT because its
initialization requires this field being initialized in JS
public const val MAX_COMPONENT_VALUE = 255

@receiver:AccessibleLateinitPropertyLiteral
KProperty0<*>.isInitialized: Boolean
    get() = throw NotImplementedError("Implementation is
intrinsic")

@receiver:AccessibleLateinitPropertyLiteral
KProperty0<*>.getValue(thisRef: Any?, property: KProperty<*>): T


that only a single thread can initialize the [Lazy] instance.

*/
    SYNCHRONIZED,

    /**
     * Initializer function can be called several times on concurrent access to uninitialized [Lazy] instance value;
     * but only the first returned value will be used as the value of [Lazy] instance value; if the instance is accessed from multiple
     * threads, its behavior is undefined.
     */
    private var _value = UNINITIALIZED_VALUE

    override val value: T
        get() {
            if (_value == UNINITIALIZED_VALUE) {
                _value = initializer!!()
                initializer = null
            }
            return _value!!
        }

    override fun isInitialized(): Boolean = _value !== UNINITIALIZED_VALUE

    override fun toString(): String = if (isInitialized())
        value.toString() else "Lazy value not initialized yet.

    private fun writeReplace(): Any =
        InitializedLazyImpl(value)
}

internal class InitializedLazyImpl<out T>(override val value: T) : Lazy<T>, Serializable {
    override fun isInitialized(): Boolean = true
    override fun toString(): String = value.toString()
}

/**
 * Counts the number of set bits in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.countOneBits(): Int

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.countLeadingZeroBits(): Int

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.countTrailingZeroBits(): Int

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.takeHighestOneBit(): Int

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.takeLowestOneBit(): Int

/**
 * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:
 * `number.rotateLeft(-n) == number.rotateRight(n)`
 * Rotating by a multiple of [Int.SIZE_BITS] (32) returns
 * the same number, or more generally
 * `number.rotateLeft(n % 32) == number.rotateLeft(n)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.rotateLeft(bitCount: Int): Int

/**
 * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
 * `number.rotateRight(-n) == number.rotateLeft(n)`
 * Rotating by a multiple of [Int.SIZE_BITS] (32) returns
 * the same number, or more generally
 * `number.rotateRight(n % 32) == number.rotateRight(n)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Int.rotateRight(bitCount: Int): Int

/**
 * Counts the number of set bits in the binary representation of this [Long] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.countOneBits(): Int

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.countLeadingZeroBits(): Int

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.countTrailingZeroBits(): Int

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.countZeroBits(): Int

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.takeHighestOneBit(): Int

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Long] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.takeLowestOneBit(): Int

/**
 * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:
 * `number.rotateLeft(-n) == number.rotateRight(n)`
 * Rotating by a multiple of [Int.SIZE_BITS] (32) returns
 * the same number, or more generally
 * `number.rotateLeft(n % 32) == number.rotateLeft(n)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateLeft(bitCount: Int): Int

/**
 * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
 * `number.rotateRight(-n) == number.rotateLeft(n)`
 * Rotating by a multiple of [Int.SIZE_BITS] (32) returns
 * the same number, or more generally
 * `number.rotateRight(n % 32) == number.rotateRight(n)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun Long.rotateRight(bitCount: Int): Int

*/
Long.countTrailingZeroBits(): Int

```
Ent

* Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,\n * or zero, if this number is zero,\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic expect fun Long.takeHighestOneBit(): Long

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Long] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic expect fun Long.takeLowestOneBit(): Long

/**
 * Rotates the binary representation of this [Long] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n *\n * number.rotateLeft(-n) == number.rotateRight(n)\n
 * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * number.rotateLeft(n % 64) == number.rotateLeft(n % 64)\n
 *An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic expect fun Long.rotateLeft(bitCount: Int): Long

/**
 * Rotates the binary representation of this [Long] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n *\n * number.rotateRight(-n) == number.rotateLeft(n)\n
 * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally\n * number.rotateRight(n % 64) == number.rotateRight(n % 64)\n
 *An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic expect fun Long.rotateRight(bitCount: Int): Long

/**
 * Counts the number of set bits in the binary representation of this [Byte] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countOneBits(): Int = (toInt() and 0xFF).countOneBits()\n
/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Byte] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countLeadingZeroBits(): Int = (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)\n
/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Byte] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits()\n
/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun Byte.takeHighestOneBit(): Byte = (toInt() and 0xFF).takeHighestOneBit().toByte()\n
/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Byte] number,\n * or zero, if this number is zero.\n*An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun Byte.takeLowestOneBit(): Byte = toInt().takeLowestOneBit().toByte()\n
/**
 * Rotates the binary representation of this [Byte] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n *\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:\n *\n * number.rotateLeft(-n) == number.rotateRight(n)\n
 * Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally\n * number.rotateLeft(n % 8) == number.rotateLeft(n % 8)\n
 *An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic fun Byte.rotateLeft(bitCount: Int): Byte =\n
} (toInt() and 0xFF).ushr(8 - (bitCount and 7)).toByte()\n
/**
 * Rotates the binary representation of this [Byte] number right by the specified [bitCount] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n *\n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:\n *\n * number.rotateRight(-n) == number.rotateLeft(n)\n
 * Rotating by a multiple of [Byte.SIZE_BITS] (8) returns the same number, or more generally\n * number.rotateRight(n % 8) == number.rotateRight(n % 8)\n
 *An@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic fun Byte.rotateRight(bitCount: Int): Byte =\n
} (toInt().shl(bitCount and 7) or (toInt() and 0xFF).ushr(8 - (bitCount and 7))).toByte()\n```
package kotlin

@file:kotlin.jvm.JvmName("TuplesKt")

import kotlin.

public data class Pair<out A, out B>(
    public val first: A,
    public val second: B
) : Serializable {

    public override fun toString(): String = "($first, $second)"
}

public infix fun <A, B>
A.to(that: B): Pair<A, B> = Pair(this, that)

/**
 * Converts this pair into a list.
 * @sample
 */
public fun <T> Pair<T, T>.toList(): List<T> = listOf(first, second)

/**
 * Represents a triad of values
 *
 * There is no meaning attached to values in this class, it can be used for any
 * purpose.
 * Triple exhibits value semantics, i.e. two triples are equal if all three components are equal.
 * An example of decomposing it into values:
 */
public data class Triple<out A, out B, out C>(
    public val first: A,
    public val second: B,
    public val third: C
) : Serializable

    /**
     * Returns string representation of the [Triple] including its [first], [second] and [third] values.
     */
    public override fun toString(): String = "($first, $second, $third)"

    companion object {
        /**
         * An empty range of values of type UInt. */
        public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)
    }

/**
 * A progression of values of type `UInt`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public open class UIntProgression
internal constructor(
    start: UInt,
    endInclusive: UInt,
    step: Int
) : Iterable<UInt>{
    init {
        if (step == 0.toInt()) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on negation.")
    }

    /**
     * The first element in the progression.
     */
    public val first: UInt = start

    /**
     * The last element in the progression.
     */
    public val last: UInt = getProgressionLastElement(start, endInclusive, step)

    /**
     * The step of the progression.
     */
    public val step: Int = step

    override fun iterator(): UIntIterator = UIntProgressionIterator(first, last, step)

    /**
     * Checks if the progression is empty.
     */
    public open fun isEmpty(): Boolean = if (step > 0) first <= last else first >= last

    override fun equals(other: Any?): Boolean =
        other is UIntProgression && (isEmpty() && other.isEmpty() ||
            first == other.first && last == other.last && step == other.step)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step.toInt())

    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"

    companion object {
        /**
         * Creates UIntProgression within the specified bounds of a closed range.
         *
         * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the
         * specified [step].
         */
        public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression = UIntProgression(rangeStart, rangeEnd, step)
    }

/**
 * An iterator over a progression of values of type `UInt`.
 * @property step the number by which the value is incremented on each step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class UIntProgressionIterator(first: UInt, last: UInt, step: Int) : UIntIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private val step = step.toUInt() // use 2-complement math for negative steps

    override fun hasNext(): Boolean = hasNext

    override fun nextUInt(): UInt {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        }
else {
    next += step
} return value
}

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin.collections

/** An iterator over a sequence of values of type `UByte`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UByteIterator : Iterator<UByte> {
    override final fun next() = nextUByte()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUByte(): UByte
}

/** An iterator over a sequence of values of type `UShort`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UShortIterator : Iterator<UShort> {
    override final fun next() = nextUShort()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUShort(): UShort
}

/** An iterator over a sequence of values of type `UInt`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UIntIterator : Iterator<UInt> {
    override final fun next() = nextUInt()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUInt(): UInt
}

/** An iterator over a sequence of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class ULongIterator : Iterator<ULong> {
    override final fun next() = nextULong()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextULong(): ULong
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// Auto-generated file. DO NOT EDIT!

package kotlin.ranges

import kotlin.internal.*

/**
 * A range of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1), ClosedRange<ULong> {
    override val start: ULong get() = first
    override val endInclusive: ULong get() = last

    override fun contains(value: ULong): Boolean = first <= value && value <= last

    override fun isEmpty(): Boolean = first > last

    override fun equals(other: Any?): Boolean =
        other is ULongRange && (isEmpty() && other.isEmpty() ||
                first == other.first && last == other.last)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())

    override fun toString(): String = "$first..$last"

    companion object {
        /**
         * An empty range of values of type ULong.
         */
        public val EMPTY: ULongRange = ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)
    }
}

/**
 * An iterator over a progression of values of type `ULong`.
 * @property step the number by which
the value is incremented on each step.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class
ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private var next = if (hasNext) first else finalElement
    override fun hasNext(): Boolean = hasNext
    override fun nextULong(): ULong {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        } else {
            next += step
        }
        return value
    }
}

/*
 * Copyright
 */

package kotlin.math

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: UInt, b: UInt): UInt {
    return minOf(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: ULong, b: ULong): ULong {
    return minOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: UInt, b: UInt): UInt {
    return maxOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: ULong, b: ULong): ULong {
    return maxOf(a, b)
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed
 * by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

@file:kotlin.jvm.JvmName("UNumbersKt")
package kotlin

/**
 * Counts the number of set bits in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countOneBits(): Int = toInt().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UInt] number,\n * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeHighestOneBit(): UInt = toInt().takeHighestOneBit().toUInt()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [UInt] number,\n * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeLowestOneBit(): UInt = toInt().takeLowestOneBit().toUInt()

/**
 * Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.\n */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateLeft(bitCount: Int): UInt = rotateLeft(bitCount).toUInt()

/**
 * Rotates the binary representation of this [UInt] number right by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateRight(bitCount: Int): UInt = rotateRight(bitCount).toUInt()
the same number, or more generally
* `number.rotateRight(n) == number.rotateRight(n % 32)`
* `number.rotateRight(n % 32)`

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt()

```

```kotlin
/**
 * Counts the number of set bits in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.countOneBits(): Int = toLong().countOneBits().toInt()\n
```

```kotlin
/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.countLeadingZeroBits(): Int = toLong().countLeadingZeroBits().toInt()\n
```

```kotlin
/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits().toInt()\n
```

```kotlin
/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.takeHighestOneBit(): ULong = toLong().takeHighestOneBit().toULong()\n
```

```kotlin
/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.takeLowestOneBit(): ULong = toLong().takeLowestOneBit().toULong()\n
```

```kotlin
/**
 * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.rotateLeft(bitCount: Int): ULong = toInt().rotateLeft(bitCount).toInt()\n
```

```kotlin
/**
 * Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULong.rotateRight(bitCount: Int): ULong = toInt().rotateRight(bitCount).toInt()\n
```

```kotlin
/**
 * Counts the number of set bits in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.countOneBits(): Int = toInt().countOneBits().toInt()\n
```

```kotlin
/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.countLeadingZeroBits(): Int = toByte().countLeadingZeroBits()\n
```

```kotlin
/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.countTrailingZeroBits(): Int = toByte().countTrailingZeroBits()\n
```

```kotlin
/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeHighestOneBit(): UByte = toInt().takeHighestOneBit().toUByte()\n
```

```kotlin
/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()\n
```
this number is zero.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()

```
@file:UseExperimental(ExperimentalUnsignedTypes::class)
package kotlin.internal

// (a - b) mod c
private fun differenceModulo(a: UInt, b: UInt, c: UInt): UInt {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

private fun differenceModulo(a: ULong, b: ULong, c: ULong): ULong {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range\n* from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative\n* [step].\n* @param param end ending bound for the progression\n* @param param step increment, or difference of successive elements in the progression\n* @return the final element of the progression\n*/
@PublishedApi
@SinceKotlin("1.3")
internal fun getProgressionLastElement(start: UByte, end: UByte, step: Byte): UByte = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toInt())
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toInt())
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Parses the string as a signed\n* number and returns the result\n* @throws IllegalFormatException if the string is not a valid
 * value for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)
representation of a number.
* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUByte(radix: Int): UByte = toUByteOrNull(radix) ?: numberFormatError(this)

* @throws NumberFormatException when the string is not a valid representation of a number.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUShort(radix: Int): UShort = toUShortOrNull(radix) ?: numberFormatError(this)

* @throws NumberFormatException when the string is not a valid representation of a number.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUInt(radix: Int): UInt = toUIntOrNull(radix) ?: numberFormatError(this)

* @throws NumberFormatException when the string is not a valid representation of a number.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toULong(radix: Int): ULong = toULongOrNull(radix) ?: numberFormatError(this)

* @throws NumberFormatException when the string is not a valid representation of a number.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUByteOrNull(): UByte? = toUByteOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUShortOrNull(): UShort? = toUShortOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUIntOrNull(): UInt? = toUIntOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toULongOrNull(): ULong? = toULongOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUByteOrNull(radix: Int): UByte? = toUByteOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUShortOrNull(radix: Int): UShort? = toUShortOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUIntOrNull(radix: Int): UInt? = toUIntOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toULongOrNull(radix: Int): ULong? = toULongOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUByteOrNull(radix: Int): UByte? = toUByteOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUShortOrNull(radix: Int): UShort? = toUShortOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUIntOrNull(radix: Int): UInt? = toUIntOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toULongOrNull(radix: Int): ULong? = toULongOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUByteOrNull(radix: Int): UByte? = toUByteOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUShortOrNull(radix: Int): UShort? = toUShortOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toUIntOrNull(radix: Int): UInt? = toUIntOrNull(radix = 10)

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @noThrows @since Kotlin("1.3") fun @ExperimentalUnsignedTypes public fun String.toULongOrNull(radix: Int): ULong? = toULongOrNull(radix = 10)
(digit < 0) return null
\n        if (result > limitBeforeMul) \n            \n            if (limitBeforeMul == limitForMaxRadix) \n                \n                result += digit.toUInt() \n                if (result > limitBeforeMul) \n                    \n                    return null
            \n        \n        result *= uradix

        val beforeAdding = result
        result += digit.toUInt() \n        if (result > limitBeforeMul) \n            \n            if (limitBeforeMul == limitForMaxRadix) \n                \n                result += digit.toUInt() \n                if (result > limitBeforeMul) \n                    \n                    return null

        return result
    }

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(): ULong? = toULongOrNull(radix = 10)

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws
 * IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(radix: Int): ULong?

{    checkRadix(radix)\n        val length = this.length \n        if (length == 0) return null \n        val limit = ULong.MAX_VALUE \n        \n        var limitBeforeMul = limit \n        val uradix = radix.toULong() \n        var result = 0uL \n        for (i in start until length) \n            \n            val digit = digitOf(this[i], radix)

            if (digit < 0) return null
            if (result > limitBeforeMul) {
                \n                if (limitBeforeMul == limitForMaxRadix) {
                    \n                    limitBeforeMul = limit / uradix

                } else {
                    \n                    return null
                }
            }

            result *= uradix

            val beforeAdding = result
            result += digit.toUInt() \n            if (result < beforeAdding) return null // overflow has happened
    }

    return result
}
Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond to the rectangular coordinates `(x, y)` by computing the arc tangent of the value `y` / `x`; the returned value is an angle in the range from `-PI` to `PI` radians. The returned value is `NaN` when `abs(x)` > `1` or `x` is `NaN`. `atan2(y, 0.0)` is `-PI/2` for `y > 0` and `+PI/2` for `y < 0`. 

The `atan2()` function computes the `atan2()` function for special cases:

- `atan2(0.0, 0.0)` is `0.0`
- `atan2(0.0, x)` is `0.0` for `x > 0` and `-PI` for `x < 0`
- `atan2(y, +Inf)` is `0.0` for `y < 0` and `-PI` for `y > 0`
- `atan2(y, -Inf)` is `-0.0` for `-PI` radians.

Special cases:

- `atan2(NaN, x)` is `NaN` when `x` is `NaN`.
- `atan2(xy)` is `NaN` when `y` is `NaN`.
- `atan2(Inf, 0.0)` is `PI/2` for `y > 0` and `-PI/2` for `y < 0`
- `atan2(Inf, x)` is `PI/2` for finite `x` and `-atan2(-Inf, x)` is `-PI/2` for finite `x`

`atan2(y, 0.0)` is `-atan2(0, y)` and `(atan2(0, y), 0)` is `NaN`.

The `atan2()` function is used in the DNAC Platform for calculating angles in various geometric computations, such as determining the direction of linear features, and for performing certain algebraic operations involving angles. Its implementation must handle special cases, such as when one or both of the input values are `NaN` or infinite, to ensure accurate results across a wide range of input values.
Computes the binary logarithm (base 2) of the value \[x\].\n
\*\An@SinceKotlin("1.2")\publicn expect fun log2(x: Double): Double\n
** Computes \(\ln(x + 1)\).\n
\*\n
This function can be implemented to produce more precise result for \([x]\) near zero.\n
\*\n
* Special cases:\n
- \(\ln1p(NaN)\) is \(NaN\)\n- \(\ln1p(x)\) is \(NaN\) where \(x < -1.0\)\n- \(\ln1p(-1.0)\) is \(-Inf\)\n- \(\ln1p(+Inf)\) is \(+Inf\)\n
* @see \([ln] function\) * @see \([expn] function\) * @\An@SinceKotlin("1.2")\publicn expect fun \(ln1p(x: Double): Double\n
\*\n
Rounds the given value \([x]\) to an integer towards positive infinity.\n
* @return the smallest double value that is greater than or equal to the given value \([x]\) and is a mathematical integer.\n
\*\n
* Special cases:\n
- `cei(x)` is \(x\) where \(x\) is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n
\*\n
\*\n
Returns the sign of the given value \([x]\):\n
- `-1.0` if the value is negative,\n- zero if the value is zero,\n- `-1.0` if the value is positive,\n- `NaN` if `NaN` is `NaN`\n
\*\n
\*\n
Returns the ulp (unit in the last place) of this value.\n
\*\n
\*\n
Returns the value nearest to this value in direction from this value towards the value \([to]\).\n
\*\n
* @return the value nearest to this value in direction from this value towards the value \([to]\).
Open Source Used In DNAC Platform Arica 12432

*/
@SinceKotlin("1.2")
public expect fun Double.nextTowards(to: Double): Double

/**
 * Rounds this [Double] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 * Rounds this [Double] value to the nearest integer and converts the result to [Long].
 * Ties are rounded towards positive infinity.
 */
@SinceKotlin("1.2")
public expect fun Double.roundToLong(): Long

/**
 * Computes the sine of the angle [x] given in radians.
 * Special cases:
 * - `sin(NaN) = NaN`
 * - `sin(+Inf)` and `-Inf)` is `0`
 */
@SinceKotlin("1.2")
public expect fun sin(x: Float): Float

/**
 * Computes the cosine of the angle [x] given in radians.
 * Special cases:
 * - `cos(NaN) = NaN`
 */
@SinceKotlin("1.2")
public expect fun cos(x: Float): Float

/**
 * Computes the tangent of the angle [x] given in radians.
 * Special cases:
 * - `tan(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun tan(x: Float): Float

/**
 * Computes the arc sine of the value [x];
 * the returned value is an angle in the range from `-PI/2` to `PI` radians.
 * Special cases:
 * - `asin(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun asin(x: Float): Float

/**
 * Computes the arc cosine of the value [x];
 * the returned value is an angle in the range from `0.0` to `PI` radians.
 * Special cases:
 * - `acos(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun acos(x: Float): Float

/**
 * Computes the arc tangent of the value [x];
 * the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 * Special cases:
 * - `atan(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun atan(x: Float): Float

/**
 * Computes the arc tangent of the value [y] / [x];
 * the returned value is an angle in the range from `-PI` to `PI` radians.
 * Special cases:
 * - `atan2(0.0, 0.0)` is `0.0`
 */
@SinceKotlin("1.2")
public expect fun atan2(y: Float, x: Float): Float

/**
 * Computes the hyperbolic sine of the value [x].
 * Special cases:
 * - `sinh(NaN) = NaN`
 */
@SinceKotlin("1.2")
public expect fun sinh(x: Float): Float

/**
 * Computes the hyperbolic cosine of the value [x].
 * Special cases:
 * - `cosh(NaN) = NaN`
 */
@SinceKotlin("1.2")
public expect fun cosh(x: Float): Float

/**
 * Computes the hyperbolic tangent of the value [x].
 * Special cases:
 * - `tanh(NaN) = NaN`
 */
@SinceKotlin("1.2")
public expect fun tanh(x: Float): Float

/**
 * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.
 * Special cases:
 * - `sqrt(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun sqrt(x: Float): Float


```kotlin
@SinceKotlin("1.2")
public expect val Float.absoluteValue: Float

/**
 * Returns the absolute value of this value:
 *   - `0` if the value is zero,
 *   - `-1.0` if the value is negative,
 *   - `1.0` if the value is positive
 *
 * Special cases:
 *   - `NaN.sign` is `NaN`
 */
@SinceKotlin("1.2")
public expect val Float.sign: Float

/**
 * Returns this value with the sign bit same as of the [sign] value:
 *   - `NaN.sign` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Float): Float

/**
 * Returns this value with the sign bit same as of the [sign] value:
 *   - `NaN.sign` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun Float.withSign(sign: Int): Float

/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 *   - `x.roundToInt()` == `Int.MAX_VALUE` when `x > Int.MAX_VALUE`
 *   - `x.roundToInt()` == `Int.MIN_VALUE` when `x < Int.MIN_VALUE`
 */
@SinceKotlin("1.2")
public expect fun Float.roundToInt(): Int

/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Long].
 * Ties are rounded towards positive infinity.
 *   - `x.roundToLong()` == `Long.MAX_VALUE` when `x > Long.MAX_VALUE`
 *   - `x.roundToLong()` == `Long.MIN_VALUE` when `x < Long.MIN_VALUE`
 */
@SinceKotlin("1.2")
public expect fun Float.roundToLong(): Long

 commentaire Integer Math
```

```kotlin
@SinceKotlin("1.2")
public expect fun abs(n: Int): Int

/**
 * Returns the absolute value of the given value 
 *   - `abs(n)` is `n` if `n` is positive,
 *   - `abs(n)` is `-n` if `n` is negative
 */
@SinceKotlin("1.2")
public expect fun abs(n: Long): Long

/**
 * Returns the absolute value of the given value 
 *   - `abs(n)` is `n` if `n` is positive,
 *   - `abs(n)` is `-n` if `n` is negative
 */
@SinceKotlin("1.2")
public expect fun abs(n: Double): Double

```
Open Source Used in DNAC Platform Articles
Open Source Used In DNAC Platform Arica 12458
Open Source Used In DNAC Platform Arica

12466
Open Source Used In DNAC Platform Arica

BhzZjB,C:MBibZZ,WAwA,iBAaA,iAAhB,C:MACX,aaAUU,CAAV,MAaKB,IAAIB,M;QACI,IAAK,WxAhIqB,GA
wP,UAaK,CAAL,CaxIO,EAnB,KawqB,CAAM,CAAN,CaxIF,CAwInB,C;:MAxIT,OA0IO,I;K1AvIX,iC;:MA
kI,WAkwB0ZO,MAAK,KBgbIZK,gKBb0ZL,ElBqzZD,KA2I3B,0kBh0ZjB,C:MBib0ZZ,WAwA,iBAaA,iAAhB,C;
MACX,aAAU,CAAV,MAaKB,IAAIB,M;QACI,IAAK,Wx9qB,GA8P,sBAaK,CAAL,EA9I0,EAaA,KB8IqB,
CAAM,CAAN,CaxIF,CaIrB,C;:MA9I0,OAgJ0,I;K8E716X,yBAAA,GE;MBk/IZA,IB;MBb+ZZ,8C;QAQI,
WKBh0ZO,MAAK,KBgsZK,gKBh0ZL,ElBgsZW,KAAM,OkBh0ZjB,C:QBlsZZ,WAwA,eAAa,IAAIhB,C:QAQX,a
AAU,CAAV,MAaKB,IAAIB,M;UACIL,AIhB,M;UACIL,IAAK,WAAU,AUAU,CAAV,EAaA,MAAAM,CAAN,CAAN
b,C:QAET,OA0AO,I;OKAbX,C;EAgBa,yB;MAAA,gE;MBk/sZ,A,IB;MBb+ZZ,8C;QAQI,WKBh0ZO,MAAK,KBgb
ZK,gKBh0ZL,ElBgqZZ,KAAM,OkBh0ZjB,C:QBlsZZ,WAwA,eAAa,IAAIhB,C:QAQX,aAAU,CAAV,MAaKB,IA
AIhB,M;UACIL,IAAK,WAAU,AUAU,CAAV,EAaA,MAAAM,CAAN,CAAN,b,C:QAET,OA0AO,I;OKAbX,C;EAg
Ba,yB;MAAA,gE;MBk/sZ,A,IB;MBb+ZZ,8C;QAQI,WKBh0ZO,MAAK,KBgbZK,gKBh0ZL,ElBgqZZ,KAAM,OkBh0ZjB,
C:QBlsZZ,WAwA,eAAa,IAAIhB,C:QAQX,aAAU,CAAV,MAaKB,IAAIhB,M;UACIL,IAAK,WAAU,AUAU,CAAV,
Open Source Used In DNAC Platform Arica 12495
Open Source Used In DNAC Platform Arica

12523
Open Source Used In DNAC Platform Arica  12561
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arica 12594
Open Source Used In DNAC Platform

Arca 12601

QA/QC,

S;UAAA,W;YAAwB,oBAAR,YAAQ,Ef44ChC,0flxCa,K8B1HgC,C;;Uf44ChC,OflxCA,K8B1HgC,C;;Uf44ChC,IAAI,OAAJ,C;YAAyB,aA

AO,K;YAAP,e;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I

A7CvD,uC;MAAA,qD;MACgC,wBAAK,eAAW,IAAX,CAAL,C;MADb,C,Y;;;;;QAC/C,aAAO,I;;;Me74CH,iB;K;oCAGJ,Y;MAAkC,OAAA,IAAK,QAAQ,OAAb,B,QAAb,B,C;K;I
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2618 base64id 1.0.0 1.0.0
1.2618.1 Available under license:
(The MIT License)

Copyright (c) 2012-2016 Kristian Faeldt <faeldt_kristian@cyberagent.co.jp>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2619 get-stream 5.1.0 5.1.0
1.2619.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2620 @babel/plugin-transform-new-target
7.16.0
1.2620.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2621 locate-path 2.0.0

1.2621.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2622 hbase-protocol 2.4.2
1.2622.1 Available under license :

Apache HBase - Protocol
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

(an example is provided in the Appendix below).
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICEx text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2623 socket.io-parser 3.3.1 3.3.1
1.2623.1 Available under license :

(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the 'Software'), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
 COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
 CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2624 global-modules 1.0.0 1.0.0

1.2624.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"); to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2625 commons-dbcp 2.6.0

1.2625.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons DBCP
Copyright 2001-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
1.2626 hadoop 3.2.2

1.2626.1 Available under license:

Copyright (c) 2003-2014 Christopher M. Kohlhoff (chris at kohlhoff dot com)

Distributed under the Boost Software License, Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt)
Use of this software is granted under one of the following two licenses, to be chosen freely by the user.


===============================================================================
Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2. The MIT License

===============================================================================
Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
     contributors may be used to endorse or promote products derived from
     this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveldb:leveldb-api (https://github.com/dain/leveldb), which has the
following notices:
   * Copyright 2011 Dain Sundstrom <dain@iq80.com>
   * Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The binary distribution of this product bundles binaries of
Gson 2.2.4,
which has the following notices:

   The Netty Project
   ------------------

Please visit the Netty web site for more information:

   * http://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/
This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
  * HOMEPAGE:
    * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
  * HOMEPAGE:
    * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
  * HOMEPAGE:
This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons Logging.txt (Apache License 2.0)
This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.alto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of 'JDOM 1.1', which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codeцу language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.
Original source copyright:Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of
Commons Lang 2.6,
which has the following notices:
* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of
Apache Log4j 1.2.17,
which has the following notices:
* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of
"Java Concurrency in Practice" book annotations 1.0,
which has the following notices:
* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5)
Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of
Jetty :: Http Utility 9.3.19.,
Jetty :: IO Utility 9.3.19.,
Jetty :: Security 9.3.19.,
Jetty :: Server Core 9.3.19.,
Jetty :: Servlet Handling 9.3.19.,
Jetty :: Utilities 9.3.19.,
Jetty :: Utilities :: Ajax,
Jetty :: Webapp Application Support 9.3.19.,
Jetty :: XML utilities 9.3.19.,
which has the following notices:
* ==================================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.
Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit.org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications
are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay
The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* == NOTICE file corresponding to section 4(d) of the Apache License, ==
  == Version 2.0, in this case for the Apache Xerces Java distribution. ==

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

--
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0
--
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting
it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was
relicensed so we could use it as Creative Commons Attribution 3.0. The license is bundled with the download available here: http://www.vectorportal.com/subcategory/205/KILLER-WHALE-FREE-VECTOR.eps/ifile/9136/detailtest.asp

--

This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

--

This product includes portions of the Guava project v14, specifically hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0, which has the following notices:

 Apache Phoenix
 Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright: Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* ==============================================================
  = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
  ==============================================================
  This product includes software developed by
  Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.
This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
uriparser - RFC 3986 URI parsing library

Copyright (C) 2007, Weijia Song <songweijia@gmail.com>
Copyright (C) 2007, Sebastian Pipping <webmaster@hartwork.org>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
* Copyright (c) 2005, European Commission project OneLab under contract 034819 (http://www.one-lab.org)
* All rights reserved.
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* - Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* - Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* - Neither the name of the University Catholique de Louvain - UCL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
http://sourceforge.net/projects/slicing-by-8/
/
*/++
*
* Copyright (c) 2004-2006 Intel Corporation - All Rights Reserved
*
* This software program is licensed subject to the BSD License,
* available at http://www.opensource.org/licenses/bsd-license.html
*
* Abstract: The main routine
*
* ...
/

For src/main/native/src/org/apache/hadoop/io/compress/lz4/[lz4.h,lz4.c,lz4hc.h,lz4hc.c],
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at:
- LZ4 source repository: http://code.google.com/p/lz4/
- LZ4 public forum: https://groups.google.com/forum/#!forum/lz4c

*/

For hadoop-common-project/hadoop-common/src/main/native/gtest

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

The binary distribution of this product bundles these dependencies under the following license:

re2j 1.1

(release 2) (Go license)

This is a work derived from Russ Cox's RE2 in Go, whose license http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h
---------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni
(https://github.com/fusesource/leveldbjni), which is available under the
following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the distribution.

* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS `AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The binary distribution of this product bundles binaries of leveldb
(http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of snappy
(http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
```
<table>
<thead>
<tr>
<th>Directory</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static</td>
<td>dataTables.bootstrap.js</td>
</tr>
<tr>
<td>hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static</td>
<td>dataTables.bootstrap.css</td>
</tr>
<tr>
<td>hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static</td>
<td>jquery.dataTables.min.js</td>
</tr>
<tr>
<td>hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static</td>
<td>dt-1.10.7/</td>
</tr>
</tbody>
</table>
```

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
```
<table>
<thead>
<tr>
<th>Directory</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static</td>
<td>dust-full-2.0.0.min.js</td>
</tr>
<tr>
<td>hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static</td>
<td>dust-helpers-1.1.1.min.js</td>
</tr>
</tbody>
</table>
```

--------------------------------------------------------------------------------------------------
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

-----------------------------------------------------------------------------------------------

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the
following license:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css
bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
e姆-helpers v0.5.13
e姆-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.13.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.0.6
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
ember-d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:
The MIT License (MIT)

Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js

The MIT License

Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery

Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
oj! Algorithms - version 43.0


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

Copyright (c) 2014 Ivan Bozhanov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:

hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:

HSQLDB Database 2.3.4

(HSQL License)

"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQLDB.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its
contributors may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the
Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the
following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of
computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with
Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to
grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this
4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial
Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.
The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the
jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
gizzly-framework 2.2.21
gizzly-http 2.2.21
gizzly-http-server 2.2.21
gizzly-http-servlet 2.2.21
gizzly-rcm 2.2.21

--------------------------------------------------------------------------------
(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or
contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification; or
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants
You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual
property rights (other than patent or trademark) Licensable by Initial
Developer, to use, reproduce, modify, display, perform, sublicense and
distribute the Original Software (or portions thereof), with or without
Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims
infringed by the making, using or selling of Original Software, to make, have
made, use, practice, sell, and offer for sale, and/or otherwise dispose of the
Original Software (or portions thereof).
(c) The licenses granted in Sections
2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms
of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or (2) for
infringements caused by: (i) the modification of the Original Software, or (ii)
the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and
subject to third party intellectual property claims, each Contributor hereby
grants You a world-wide, royalty-free, non-exclusive license:
(a) under
intellectual property rights (other than patent or trademark) Licensable by
Contributor to use, reproduce, modify, display, perform, sublicense and
distribute the Modifications created by such Contributor (or portions thereof),
either on an unmodified basis, with other Modifications, as Covered Software
and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that Contributor either alone
and/or in combination with its Contributor Version (or portions of such
combination), to make, use, sell, offer for sale, have made, and/or otherwise
dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such...
Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND
8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0
This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  `src/google/protobuf/stubs/atomicops_internals_generic_gcc.h`.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  `src/google/protobuf/stubs/atomicops_internals_power.h`.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

By obtaining, using and/or copying this work, you (the licensee) agree that you
have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court
requires any other Contributor to pay any damages as a result, the Commercial
Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE,
NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each
Recipient is solely responsible for determining the appropriateness of using and
distributing the Program and assumes all risks associated with its exercise of
rights under this Agreement, including but not limited to the risks and costs
of program errors, compliance with applicable laws, damage to or loss of data,
programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRict LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable
law, it shall not affect the validity or enforceability of the remainder of the
terms of this Agreement, and without further action by the parties hereto, such
provision shall be reformed to the minimum extent necessary to make such
provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware)
infringes such Recipient's patent(s), then such Recipient's rights granted under
Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to
comply with any of the material terms or conditions of this Agreement and does
not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement
and any licenses granted by Recipient relating to the Program shall continue and
survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

(3-clause BSD)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
FindBugs-jsr305 3.0.0
dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

--------------------------------------------------------------------------------
(2-clause BSD)
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those
of the authors and should not be interpreted as representing official policies,
either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the
following license:
"Java Concurrency in Practice" book annotations 1.0

--------------------------------------------------------------------------------
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS
PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR
OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as
incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and
perform publicly by means of a digital audio transmission Derivative Works.
For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive
right to collect, whether individually or via a performance rights society (e.g.
ASCAP, BMI, SESAC), royalties for the public performance or public digital
performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right
to collect, whether individually or via a music rights agency or designated
agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the
Work ("cover version") and distribute, subject to the compulsory license created
by 17 USC Section 115 of the US Copyright Act (or the equivalent in other
jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the
Work is a sound recording, Licensor waives the exclusive right to collect,
whether individually or via a performance-rights society (e.g. SoundExchange),
royalties for the public digital performance (e.g. webcast) of the Work, subject
to the compulsory license created by 17 USC Section 114 of the US Copyright Act
(or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or
hereafter devised. The above rights include the right to make such modifications
as are technically necessary to exercise the rights in other media and formats.
All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject
to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally
perform the Work only under the terms of this License, and You must include a
copy of, or the Uniform Resource Identifier for, this License with every copy or
phonorecord of the Work You distribute, publicly display, publicly perform, or
publicly digitally perform. You may not offer or impose any terms on the Work
that alter or restrict the terms of this License or the recipients' exercise of
the rights granted hereunder. You may not sublicense the Work. You must keep
intact all notices that refer to this License and to the disclaimer of
warranties. You may not distribute, publicly display, publicly perform, or
publicly digitally perform the Work with any technological measures that control
access or use of the Work in a manner inconsistent with the terms of this
License Agreement. The above applies to the Work as incorporated in a Collective
Work, but this does not require the Collective Work apart from the Work itself
to be made subject to the terms of this License. If You create a Collective
Work, upon notice from any Licensor You must, to the extent practicable, remove
from the Collective Work any credit as required by clause 4(b), as requested. If
You create a Derivative Work, upon notice from any Licensor You must, to the
extent practicable, remove from the Derivative Work any credit as required by
clause 4(b), as requested.
If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor’s copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any
such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

jamon-runtime 2.4.1

--------------------------------------------------------------------------------
(MPL 2.0)
Mozilla Public License
Version 2.0

1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.
1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.
1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For legal entities, You includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

for any code that a Contributor has removed from Covered Software; or

for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a
timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent
prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character
including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).
10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:
JDOM 1.1

FLICTEDLicenses Notice

/*--

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

Open Source Used In DNAC Platform Arica 12686
4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org/)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the following license:

Hbase Server 1.2.4

This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS
INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work,
the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.
2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

   a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
   b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish." or a modification could indicate "The original work has been modified.";
   c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
   d. to Distribute and Publicly Perform Adaptations.
   e. For the avoidance of doubt:
      i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
      ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
      iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:
a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(b), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this
License. You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.
Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2010 Ben Noordhuis

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2013, Michael Bostock
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name Michael Bostock may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL MICHAEL BOSTOCK BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2627 gensync 1.0.0-beta.2
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2016, the original author or authors.
* 
* This software is distributable under the BSD license. See the terms of the 
* BSD license in the documentation provided with this software.
* 
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/StringsCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/package-info.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Binding.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/DefaultHighlighter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Reference.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/NullCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/AggregateCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Parser.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/ReaderUtils.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/history/package-info.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/highlighter/package-info.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Highlighter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Widget.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/DefaultExpander.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/UndoTree.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/EndOfIOException.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/UserInterruptedException.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/EnumCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Expander.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Macro.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/keymap/KeyMap.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2019, the original author or authors.
* 
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
*
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/LineReaderImpl.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2017, the original author or authors.
*
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
*
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/BufferImpl.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/SimpleMaskingCallback.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Buffer.java
No license file was found, but licenses were detected in source scan.

2016, the original author or authors.

This software is distributable under the BSD license. See the terms of the
BSD license in the documentation provided with this software.

https://opensource.org/licenses/BSD

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/META-INF/maven/org.jline/jline-reader/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/EOFError.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/SyntaxError.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2018, the original author or authors.
* 
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* 
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/DefaultParser.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/LineReader.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/MaskingCallback.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/LineReaderBuilder.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/history/DefaultHistory.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/Completer.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/ArgumentCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/impl/completer/FileNameCompleter.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/ParsedLine.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/keymap/BindingReader.java
* /opt/cola/permits/1114600224_1607534234.11/0/jline-reader-3-12-1-sources-jar/org/jline/reader/CompletingParsedLine.java
1.2630 wrap-ansi 6.2.0 6.2.0
1.2630.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2631 odata-parser 1.4.1 1.4.1
1.2631.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Auth0, Inc. <support@auth0.com> (http://auth0.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
SOFTWARE.

1.2632 expat 2.1.0

1.2632.1 Available under license:
Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper
Copyright (c) 2001-2019 Expat maintainers

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2633 getpass 0.1.7 0.1.7

1.2633.1 Available under license:
Copyright Joyent, Inc. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
1.2634 types-color-name 1.1.1
1.2634.1 Available under license:
   MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE

1.2635 kite-data-core 1.1.0
1.2635.1 Available under license:
   No license file was found, but licenses were detected in source scan.

#
# Copyright 2013 Cloudera Inc.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
org.kitesdk.data.spi.filesystem.Loader

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/META-INF/services/org.kitesdk.data.spi.Loadable
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 Cloudera.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/StorageKey.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/View.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/MultiLevelIterator.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/AbstractDatasetRepository.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/AbstractMetadataProvider.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/Marker.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/AbstractRefinableView.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/UnknownFormatException.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/MarkerRange.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/URIPattern.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/FileSystemPartitionIterator.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 joey.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1.jar/org/kitesdk/data/spi/DatasetRepositories.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1.jar/org/kitesdk/data/CompressionType.java

No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1.jar/org/kitesdk/data/spi/filesystem/AvroFSInput.java
   No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2015 Cloudera Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1.jar/org/kitesdk/data/spi/filesystem/SignalManager.java
   No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 Cloudera.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1.jar/org/kitesdk/data/Signalable.java
   No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2014 Cloudera, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/DataModelUtil.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 Cloudera Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/FilteredReader.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/PartitionStrategyParser.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/ColumnMappingParser.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/FileSystemViewKeyInputFormat.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/DescriptorUtil.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2015 Cloudera Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/PartitionView.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/Replaceable.java

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2013 Cloudera Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/DatasetException.java

No license file was found, but licenses were detected in source scan.

/**
* Copyright 2013 Cloudera Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/filesystem/MultiFileDatasetReader.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/SchemaValidationUtil.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/filesystem/ParquetAppender.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/package-info.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/SchemaNotFoundException.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/partition/IntRangeFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/partition/HashFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/partition/MinuteFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/partition/PartitionListener.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/partition/PartitionFunctions.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/ColumnMapping.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/FieldMapping.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/ListFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/spi/DayOfMonthFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/Accessor.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/ColumnMapping.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/FieldMapping.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/ListFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/DayOfMonthFieldPartitioner.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
  jar/org/kitesdk/data/impl/Accessor.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Cloudera Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
 * Copyright 2013 Cloudera Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * Copyright 2013 Cloudera Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/PartitionedDataset.java

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/FileSystemUtil.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/TemporaryDatasetRepositoryAccessor.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/DatasetRecordException.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/CSVUtil.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/CSVAppender.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/TemporaryFileSystemDatasetRepository.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/SchemaUtil.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/Pair.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/Flushable.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/JSONFileReader.java
* /opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/filesystem/CSVRecordBuilder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 Cloudera.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1202446424_1631568199.51/0/kite-data-core-1-1-0-sources-1-jar/org/kitesdk/data/spi/Conversions.java
1.2636 snake-yaml 1.29
1.2636.1 Available under license:

Found license 'GNU Lesser General Public License' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

Found license 'Eclipse Public License 1.0' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal // LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html // GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html // AL, Apache License, V2.0 or later, http://www.apache.org/licenses // BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

1.2637 camelcase 5.0.0 5.0.0
1.2637.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2638 jackson-databind 2.11.4
1.2638.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.
You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.2639 babel-plugin-transform-exponentiation-operator 6.24.1 6.24.1

1.2639.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2640 mongodb-driver-sync 4.2.3

1.2640.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2008-present MongoDB, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/DistinctIterableImpl.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/ChangeStreamIterableImpl.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/KeyManagementService.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/MongoCursor.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/MongoCursor.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/ListIndexesIterableImpl.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/MongoDatabase.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/package-info.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/OperationExecutor.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/gridfs/GridFSUploadStreamImpl.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/MongoDatabase.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
jar/com/mongodb/client/internal/MappingIterable.java
* /opt/cola/permits/1202446882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources-
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2008-present MongoDB, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

**Found in path(s):**
* /opt/cola/permits/120246882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources.jar/com/mongodb/client/ClientSession.java
* /opt/cola/permits/120246882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources.jar/com/mongodb/client/internal/package-info.java
* /opt/cola/permits/120246882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources.jar/com/mongodb/client/TransactionBody.java
* /opt/cola/permits/120246882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources.jar/com/mongodb/client/internal/ClientSessionClock.java
* /opt/cola/permits/120246882_1631568039.69/0/mongodb-driver-sync-4-2-3-sources.jar/com/mongodb/client/internal/ClientSessionImpl.java

### 1.2641 graceful-fs 4.2.4 4.2.4

#### 1.2641.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter, Ben Noordhuis, and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2642 okhttp 4.8.0

1.2642.1 Available under license:
Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

1.2643 https-proxy-agent 3.0.1 3.0.1

1.2643.1 Available under license:
No license file was found, but licenses were detected in source scan.

https-proxy-agent
================
### An HTTP(s) proxy `http.Agent` implementation for HTTPS

This module provides an `http.Agent` implementation that connects to a specified
HTTP or HTTPS proxy server, and can be used with the built-in `https` module.

Specifically, this `Agent` implementation connects to an intermediary "proxy"
server and issues the [CONNECT HTTP method][CONNECT], which tells the proxy to
open a direct TCP connection to the destination server.

Since this agent implements the CONNECT HTTP method, it also works with other
protocols that use this method when connecting over proxies (i.e. WebSockets).
See the "Examples" section below for more.

Installation
------------

Install with `npm`:

```
```bash
$ npm install https-proxy-agent
```

Examples
--------
### `https` module example

```js
```
var url = require('url');
var https = require('https');
var HttpsProxyAgent = require('https-proxy-agent');

// HTTP/HTTPS proxy to connect to
var proxy = process.env.http_proxy || 'http://168.63.76.32:3128';
console.log('using proxy server %j', proxy);

// HTTPS endpoint for the proxy to connect to
console.log('attempting to GET %j', endpoint);
var options = url.parse(endpoint);

// create an instance of the `HttpsProxyAgent` class with the proxy server information
var agent = new HttpsProxyAgent(proxy);
options.agent = agent;

https.get(options, function (res) {
  console.log("response" event!, res.headers);
  res.pipe(process.stdout);
});
```
var socket = new WebSocket(endpoint, { agent: agent });

socket.on('open', function () {
  console.log("open" event!);
  socket.send('hello world');
});

socket.on('message', function (data, flags) {
  console.log("message" event! %j %j", data, flags);
  socket.close();
});
```

API
---

### new HttpsProxyAgent(Object options)

The `HttpsProxyAgent` class implements an `http.Agent` subclass that connects to the specified "HTTP(s) proxy server" in order to proxy HTTPS and/or WebSocket requests. This is achieved by using the [HTTP `CONNECT` method][CONNECT].

The `options` argument may either be a string URI of the proxy server to use, or an "options" object with more specific properties:

- *host* - String - Proxy host to connect to (may use `hostname` as well). Required.
- *port* - Number - Proxy port to connect to. Required.
- *protocol* - String - If `https:`, then use TLS to connect to the proxy.
- *headers* - Object - Additional HTTP headers to be sent on the HTTP CONNECT method.
- Any other options given are passed to the `net.connect()`/`tls.connect()` functions.

License
------

(The MIT License)

Copyright (c) 2013 Nathan Rajlich &lt;nathan@tootallnate.net&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

[CONNECT]: http://en.wikipedia.org/wiki/HTTP_tunnel#HTTP_CONNECT_Tunneling

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1087657923_1599578708.88/0/https-proxy-agent-3-0-1-2-tgz/package/README.md

1.2644 twill-zookeeper 0.6.0-incubating
1.2644.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/BasicACLData.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/AbstractZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/ACLData.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/zookeeper/ReentrantDistributedLock.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/ForwardingZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/zookeeper/BasicNodeChildren.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/OperationFuture.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/zookeeper/SettableOperationFuture.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/NamespaceZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/ZKClients.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/zookeeper/InMemoryZKServer.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/ZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/zookeeper/package-info.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/zookeeper/BasicNodeData.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/KillZKSession.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/LeaderElection.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/NodeData.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/RetryUtils.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/package-info.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/RewatchOnExpireWatcher.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/RewatchOnExpireWatcher.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/FailureRetryZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/RetryStrategies.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/NodeChildren.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/ZKClientService.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/ZKOperations.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/zookeeper/RewatchOnExpireZKClient.java
* /opt/cola/permits/1137746061_1614121450.21/0/twill-zookeeper-0-6-0-incubating-sources-2-jar/org/apache/twill/zookeeper/RetryStrategy.java

1.2645 jsreport-templates 2.4.0 2.4.0
1.2645.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

Open Source Used In DNAC Platform Arica 12726
"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.


You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying
Corresponding Source.

1) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer
system, and (b) will operate properly with a modified version
of the Library that is interface-compatible with the Linked
Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2646 libpciaccess 0.13.4 1

1.2646.1 Available under license:

(C) Copyright IBM Corporation 2006, 2007
(C) Copyright Eric Anholt 2006
(C) Copyright Mark Kettenis 2011
(C) Copyright Robert Millan 2012
Copyright 2009, 2012 Red Hat, Inc.
All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation on the rights to use, copy, modify, merge, publish, distribute, sub license, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL IBM AND/OR THEIR SUPPLIERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

-----------------------------------------------------------------------------------

Copyright (c) 2008 Juan Romero Pardines
Copyright (c) 2008, 2011 Mark Kettenis
Copyright (c) 2009 Michael Lorenz
Copyright (c) 2009, 2012 Samuel Thibault

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

------------------------------------------------------------------------------

Copyright (C) 2000 The XFree86 Project, Inc. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
XFREE86 PROJECT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the XFree86 Project shall
not be used in advertising or otherwise to promote the sale, use or other
dealings in this Software without prior written authorization from the
XFree86 Project.

------------------------------------------------------------------------------

Copyright (c) 2007 Paulo R. Zanoni, Tiago Vignatti
Copyright (c) 2009 Tiago Vignatti

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2647 run-async 2.4.1

1.2647.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Simon Boudrias

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2648 expand-range 1.8.2

1.2648.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2649 @babel/helper-replace-supers 7.16.0
1.2649.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2650 browserslist 3.2.8 3.2.8

1.2650.1 Available under license :
The MIT License (MIT)

Copyright 2014 Andrey Sitnik <andrey@sitnik.ru>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2651 hasha 5.2.2

1.2651.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (https://sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2652 registry-url 5.1.0

1.2652.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2653 validatejs 0.12.0 0.12.0

1.2653.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013-2016 Nicklas Ansman, 2013 Wrapp

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
THE SOFTWARE.

1.2654 streams 2.7.0

1.2655 form-data 2.5.1 2.5.1
1.2655.1 Available under license :

Copyright (c) 2012 Felix Geisendrfer (felix@debuggable.com) and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2656 @babel/helper-validator-identifier

7.10.4 7.10.4
1.2656.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2657 component-emitter 1.2.1 1.2.1

1.2657.1 Available under license:
(The MIT License)

Copyright (c) 2014 Component contributors <dev@component.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2658 array-unique 0.3.2

1.2658.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2659 y18n 3.2.1 3.2.1
1.2659.1 Available under license :

Copyright (c) 2015, Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2660 @babel/plugin-proposal-unicode-property-regex 7.16.0
1.2660.1 Available under license :

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2661 lodash-flatten 4.4.0 4.4.0

1.2661.1 Available under license:
Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2662 sqlite-jdbc 3.30.1

1.2662.1 Available under license:

Copyright (c) 2006, David Crawshaw. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2663 hadoop-auth 3.2.2
1.2663.1 Available under license :

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).

The binary distribution of this product bundles binaries of
org.iq80.leveledb:leveldb-api (https://github.com/dain/leveldb), which has the
following notices:
* Copyright 2011 Dain Sundstrom <dain@iq80.com>
* Copyright 2011 FuseSource Corp. http://fusesource.com

The binary distribution of this product bundles binaries of
AWS SDK for Java - Bundle 1.11.563 (https://github.com/aws/aws-sdk-java),
which has the following notices:

This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-
2010 AOL Inc.

The binary distribution of this product bundles binaries of
Gson 2.2.4,
which has the following notices:

The Netty Project

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2014 The Netty Project
The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

------------------------------------------------------------------------

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
This product contains a modified portion of 'ArrayDeque', written by Josh Bloch of Google, Inc:

* LICENSE:
  * license/LICENSE.deque.txt (Public Domain)

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* LICENSE:
  * license/LICENSE.harmony.txt (Apache License 2.0)
  * HOMEPAGE:
    * http://archive.apache.org/dist/harmony/

This product contains a modified version of Roland Kuhn's ASL2 AbstractNodeQueue, which is based on Dmitriy Vyukov's non-intrusive MPSC queue. It can be obtained at:

* LICENSE:
  * license/LICENSE.abstractnodequeue.txt (Public Domain)
  * HOMEPAGE:
    * https://github.com/akka/akka/blob/wip-2.2.3-for-scala-2.11/akka-actor/src/main/java/akka/dispatch/AbstractNodeQueue.java

This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
  * HOMEPAGE:
    * https://code.google.com/p/libdivsufsort/

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
* HOMEPAGE:
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
Open Source Used In DNAC Platform Arica 12749

* HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
  * HOMEPAGE:
    * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
  * HOMEPAGE:
    * http://code.google.com/p/snappy/

This product contains a modified portion of UnsignedBytes LexicographicalComparator from Guava v21 project by Google Inc, which can be obtained at:

* LICENSE:
  * license/COPYING (Apache License 2.0)
  * HOMEPAGE:
    * https://github.com/google/guava

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (GNU LGPL 2.1)
  * HOMEPAGE:
    * http://www.jboss.org/jbossmarshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
  * HOMEPAGE:
    * http://code.google.com/p/caliper/

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
This product optionally depends on `Apache Log4J`, a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on `Aalto XML`, an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.aalto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of `HPACK`, a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack

This product contains a modified portion of `Apache Commons Lang`, a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains a modified portion of `JDOM 1.1`, which can be obtained at:

* LICENSE:
  * https://github.com/hunterhacker/jdom/blob/jdom-1.1/core/LICENSE.txt
* HOMEPAGE:
  * http://www.jdom.org/

The binary distribution of this product bundles binaries of Commons Codec 1.4, which has the following notices:

* src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors.

Original source copyright: Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

The binary distribution of this product bundles binaries of Commons Lang 2.6, which has the following notices:

* This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

The binary distribution of this product bundles binaries of Apache Log4j 1.2.17, which has the following notices:

* ResolverUtil.java
  Copyright 2005-2006 Tim Fennell
Dumbster SMTP test server
  Copyright 2004 Jason Paul Kitchen
TypeUtil.java
  Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

The binary distribution of this product bundles binaries of "Java Concurrency in Practice" book annotations 1.0, which has the following notices:

* Copyright (c) 2005 Brian Goetz and Tim Peierls Released under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.5) Official home: http://www.jcip.net Any republication or derived work distributed in source code form must include this copyright and license notice.

The binary distribution of this product bundles binaries of Jetty :: Http Utility 9.3.19., Jetty :: IO Utility 9.3.19., Jetty :: Security 9.3.19., Jetty :: Server Core 9.3.19., Jetty :: Servlet Handling 9.3.19., Jetty :: Utilities 9.3.19., Jetty :: Utilities :: Ajax, Jetty :: Webapp Application Support 9.3.19., Jetty :: XML utilities 9.3.19., which has the following notices:

* ==============================================================
  Jetty Web Container
  Copyright 1995-2016 Mort Bay Consulting Pty Ltd.
  ==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL:
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2:
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0:
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following
artifacts may be included in the distribution or downloaded when ALPN
module is selected.

* java.sun.security.ssl
These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------

Mortbay
The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------

Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies./

The binary distribution of this product bundles binaries of Snappy for Java 1.0.4.1, which has the following notices:
* This product includes software developed by Google
  Snappy: http://code.google.com/p/snappy/ (New BSD License)

This product includes software developed by Apache
(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by "GCC RUNtime Library Exception"

== Contributors ==
* Tatu Saloranta
  * Providing benchmark suite
* Alec Wysoker
  * Performance and memory usage improvement

The binary distribution of this product bundles binaries of Xerces2 Java Parser 2.9.1, which has the following notices:
* === NOTICE file corresponding to section 4(d) of the Apache License, ===
  Version 2.0, in this case for the Apache Xerces Java distribution. ===
  =========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation
This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

The binary distribution of this product bundles binaries of
Logback Classic Module 1.1.2,
Logback Core Module 1.1.2,
which has the following notices:
* Logback: the reliable, generic, fast and flexible logging framework.
  Copyright (C) 1999-2012, QOS.ch. All rights reserved.

The binary distribution of this product bundles binaries of
Apache HBase - Annotations 1.2.6,
Apache HBase - Client 1.2.6,
Apache HBase - Common 1.2.6,
Apache HBase - Hadoop Compatibility 1.2.6,
Apache HBase - Hadoop Two Compatibility 1.2.6,
Apache HBase - Prefix Tree 1.2.6,
Apache HBase - Procedure 1.2.6,
Apache HBase - Protocol 1.2.6,
Apache HBase - Server 1.2.6,
which has the following notices:
* Apache HBase
  Copyright 2007-2015 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

---
This product incorporates portions of the 'Hadoop' project

Copyright 2007-2009 The Apache Software Foundation

Licensed under the Apache License v2.0

---
Our Orca logo we got here: http://www.vectorfree.com/jumping-orca
It is licensed Creative Commons Attribution 3.0.
See https://creativecommons.org/licenses/by/3.0/us/
We changed the logo by stripping the colored background, inverting it and then rotating it some.

Later we found that vectorfree.com image is not properly licensed.
The original is owned by vectorportal.com. The original was relicensed so we could use it as Creative Commons Attribution 3.0. The license is bundled with the download available here:

This product includes portions of the Bootstrap project v3.0.0

Copyright 2013 Twitter, Inc.

Licensed under the Apache License v2.0

This product uses the Glyphicons Halflings icon set.

http://glyphicons.com/

Copyright Jan Kovak

Licensed under the Apache License v2.0 as a part of the Bootstrap project.

This product includes portions of the Guava project v14, specifically 'hbase-common/src/main/java/org/apache/hadoop/hbase/io/LimitInputStream.java'

Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0

The binary distribution of this product bundles binaries of Phoenix Core 4.7.0, which has the following notices:
Apache Phoenix
Copyright 2013-2016 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This also includes:

The phoenix-spark module has been adapted from the phoenix-spark library distributed under the terms of the Apache 2 license. Original source copyright:
Copyright 2014 Simply Measured, Inc.
Copyright 2015 Interset Software Inc.

The file bin/daemon.py is based on the file of the same name in python-daemon 2.0.5 (https://pypi.python.org/pypi/python-daemon/). Original source copyright:
# Copyright 20082015 Ben Finney <ben+python@benfinney.id.au>
# Copyright 20072008 Robert Niederreiter, Jens Klein
# Copyright 20042005 Chad J. Schroeder
# Copyright 2003 Clark Evans
# Copyright 2002 Noah Spurrier
# Copyright 2001 Jrgen Hermann

The binary distribution of this product bundles binaries of
Plexus Cipher: encryption/decryption Component 1.4,
which has the following notices:
* The code in this component contains a class - Base64 taken from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/Base64.java
  which is Apache license: http://www.apache.org/licenses/LICENSE-2.0

The PBE key processing routine PBECipher.createCipher() is adopted from http://juliusdavies.ca/svn/not-yet-commons-ssl/tags/commons-ssl-0.3.10/src/java/org/apache/commons/ssl/OpenSSL.java
  which is also Apache APL-2.0 license: http://www.apache.org/licenses/LICENSE-2.0

The binary distribution of this product bundles binaries of
software.amazon.ion:ion-java 1.0.1,
which has the following notices:
* Amazon Ion Java Copyright 2007-2016 Amazon.com, Inc. or its affiliates. All Rights Reserved.

The binary distribution of this product bundles binaries of
joda-time:joda-time:2.9.9
which has the following notices:
* = NOTICE file corresponding to section 4d of the Apache License Version 2.0 =

This product includes software developed by
Joda.org (http://www.joda.org/).

The binary distribution of this product bundles binaries of
Ehcache 3.3.1,
which has the following notices:
* Ehcache V3 Copyright 2014-2016 Terracotta, Inc.

The binary distribution of this product bundles binaries of
snakeyaml (https://bitbucket.org/asomov/snakeyaml),
which has the following notices:
* Copyright (c) 2008, http://www.snakeyaml.org

The binary distribution of this product bundles binaries of
swagger-annotations (https://github.com/swagger-api/swagger-core),
which has the following notices:
* Copyright 2016 SmartBear Software

The binary distribution of this product bundles binaries of
metrics-core 3.2.4
which has the following notices:
* Copyright 2010-2013 Coda Hale and Yammer, Inc.
This product includes software developed by Coda Hale and Yammer, Inc.

This product includes code derived from the JSR-166 project (ThreadLocalRandom, Striped64, LongAdder), which was released with the following comments:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE HADOOP SUBCOMPONENTS:

The Apache Hadoop project contains subcomponents with separate copyright notices and license terms. Your use of the source code for the these subcomponents is subject to the terms and conditions of the following licenses.

For the org.apache.hadoop.util.bloom.* classes:

/**
 *
 * Copyright (c) 2005, European Commission project OneLab under contract
 * 034819 (http://www.one-lab.org)
 * All rights reserved.
 * Redistribution and use in source and binary forms, with or
 * without modification, are permitted provided that the following
 * conditions are met:
 * - Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * - Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the distribution.
 * - Neither the name of the University Catholique de Louvain - UCL
 * nor the names of its contributors may be used to endorse or
 * promote products derived from this software without specific prior
 * written permission.
 */
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
  * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
  * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
  * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
  * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
  * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
  * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
  * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
  * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
  * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
  * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
  * POSSIBILITY OF SUCH DAMAGE.
*/

For portions of the native implementation of slicing-by-8 CRC calculation in src/main/native/src/org/apache/hadoop/util:

Copyright (c) 2008,2009,2010 Massachusetts Institute of Technology.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Massachusetts Institute of Technology nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Other portions are under the same license from Intel:
For src/main/native/src/org/apache/hadoop/io/compress/lz4/{lz4.h,lz4.c,lz4hc.h,lz4hc.c},

/*
LZ4 - Fast LZ compression algorithm
Header File
Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at :
- LZ4 source repository : http://code.google.com/p/lz4/
- LZ4 public forum : https://groups.google.com/forum/#!forum/lz4c
*/
For hadoop-common-project/hadoop-common/src/main/native/gtest

---------------------------------------------------------------------
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the
following license:
re2j 1.1

---------------------------------------------------------------------

(GO license)
This is a work derived from Russ Cox's RE2 in Go, whose license
http://golang.org/LICENSE is as follows:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-hdfs-project/hadoop-hdfs-native-client/src/main/native/fuse-dfs/util/tree.h

---------------------------------------------------------------------
Copyright 2002 Niels Provos <provos@citi.umich.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles binaries of leveldbjni
Open Source Used In DNAC Platform Arica 12767

(https://github.com/fusesource/leveldbjni), which is available under the following license:

Copyright (c) 2011 FuseSource Corp. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of FuseSource Corp. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-nodemanager/src/main/native/container-executor/impl/compat/{fstatat|openat|unlinkat}.h:

Copyright (c) 2012 The FreeBSD Foundation
All rights reserved.

This software was developed by Pawel Jakub Dawidek under sponsorship from the FreeBSD Foundation.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=============

The binary distribution of this product bundles binaries of leveldb (http://code.google.com/p/leveldb/), which is available under the following license:

Copyright (c) 2011 The LevelDB Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The binary distribution of this product bundles binaries of snappy
(http://code.google.com/p/snappy/), which is available under the following license:

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For:
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dataTables.bootstrap.css
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery.dataTables.min.js
hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/dt-1.10.7/

--------------------------------------------------------------------------------

Copyright (C) 2008-2016, SpryMedia Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-full-2.0.0.min.js
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/dust-helpers-1.1.1.min.js

Copyright (c) 2010 Aleksander Williams

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/moment.min.js

Copyright (c) 2011-2016 Tim Wood, Iskren Chernev, Moment.js contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The binary distribution of this product bundles these dependencies under the following license:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/bootstrap-3.3.7
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/bootstrap.min.js
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap.min.css
hadoop-tools/hadoop-sls/src/main/html/css/bootstrap-responsive.min.css

bootstrap v3.3.6
broccoli-asset-rev v2.4.2
broccoli-funnel v1.0.1
datatables v1.10.8
et-helpers v0.5.13
em-table v0.1.6
ember v2.2.0
ember-array-contains-helper v1.0.2
ember-bootstrap v0.5.1
ember-cli v1.3.13
ember-cli-app-version v1.0.0
ember-cli-babel v5.1.6
ember-cli-content-security-policy v0.4.0
ember-cli-dependency-checker v1.2.0
ember-cli-htmlbars v1.0.2
ember-cli-htmlbars-inline-precompile v0.3.1
ember-cli-ic-ajax v0.2.1
ember-cli-inject-live-reload v1.4.0
ember-cli-jquery-ui v0.0.20
ember-cli-qunit v1.2.1
ember-cli-release v0.2.8
ember-cli-shims v0.6.0
ember-cli-sri v1.2.1
ember-cli-test-loader v0.2.1
ember-cli-uglify v1.2.0
d3 v0.1.0
ember-data v2.1.0
ember-disable-proxy-controllers v1.0.1
ember-export-application-global v1.0.5
ember-load-initializers v0.1.7
ember-qunit v0.4.16
ember-qunit-notifications v0.1.0
ember-resolver v2.0.3
ember-spin-spinner v0.2.3
ember-truth-helpers v1.2.0
jquery v2.1.4
jquery-ui v1.11.4
loader.js v3.3.0
momentjs v2.10.6
qunit v1.19.0
select2 v4.0.0
snippet-ss v1.11.0
spin.js v2.3.2
Azure Data Lake Store - Java client SDK 2.0.11
JCodings 1.0.8
Joni 2.1.2
Mockito 1.8.5
JUL to SLF4J bridge 1.7.25
SLF4J API Module 1.7.25
SLF4J LOG4J-12 Binding 1.7.25

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:
./hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/nvd3-1.8.5.* (css and js files)

Copyright (c) 2011-2014 Novus Partners, Inc.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-nvd3-1.0.9.min.js
--------------------------------------------------------------------------------

The MIT License (MIT)
Copyright (c) 2014 Konstantin Skipor

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-1.6.4.min.js
hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/angular-route-1.6.4.min.js
--------------------------------------------------------------------------------

The MIT License
Copyright (c) 2010-2017 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

For:
- hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/jquery-3.3.1.min.js
- hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/jquery.js
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jquery
- Apache HBase - Server which contains JQuery minified javascript library version 1.8.3
- Microsoft JDBC Driver for SQLServer - version 6.2.1.jre7

MIT License

Copyright (c) 2003-2017 Optimatika

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
For:

oj! Algorithms - version 43.0

--------------------------------------------------------------------------------


This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/jquery/jquery

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

For:

hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/webapps/static/jt/jquery.jstree.js

--------------------------------------------------------------------------------

Copyright (c) 2014 Ivan Bozhanov
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For:

hadoop-hdfs-project/hadoop-hdfs/src/main/webapps/static/d3-3.5.17.min.js

D3 is available under a 3-clause BSD license. For details, see:
hadoop-tools/hadoop-sls/src/main/html/js/thirdparty/d3-LICENSE

The binary distribution of this product bundles these dependencies under the following license:
HSQldb Database 2.3.4

(HSQL License)
"COPYRIGHTS AND LICENSES (based on BSD License)

For work developed by the HSQL Development Group:

Copyright (c) 2001-2016, The HSQL Development Group
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For work originally developed by the Hypersonic SQL Group:

Copyright (c) 1995-2000 by the Hypersonic SQL Group.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Hypersonic SQL Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Hypersonic SQL Group."

The binary distribution of this product bundles these dependencies under the following license:
Java Servlet API 3.1.0
servlet-api 2.5
jsp-api 2.1
jsr311-api 1.1.1
Glassfish Jasper 6.1.14
Servlet Specification 2.5 API 6.1.14

(CDDL 1.0)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1.Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2.Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3.Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4.Executable means the Covered Software in any form other than Source Code.

1.5.Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6.Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7.License means this document.

1.8.Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights
conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions thereof), either on an unmodified basis, with other Modifications, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source
Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.
3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a)rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b)otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.
6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer
software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.
As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
jersey-client 1.19
jersey-core 1.19
jersey-grizzly2 1.19
jersey-grizzly2-servlet 1.19
jersey-json 1.19
jersey-server 1.19
jersey-servlet 1.19
jersey-guice 1.19
Jersey Test Framework - Grizzly 2 Module 1.19
JAXB RI 2.2.3
Java Architecture for XML Binding 2.2.11
grizzly-framework 2.2.21
grizzly-http 2.2.21
grizzly-http-server 2.2.21
grizzly-http-servlet 2.2.21
grizzly-rcm 2.2.21

(CDDL 1.1)
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. Executable means the Covered Software in any form other than Source Code.
1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. License means this document.
1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. Modifications means the Source Code and Executable form of any of the following:
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   B. Any new file that contains any part of the Original Software or previous Modification;
   C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal
entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof);
and (2) the combination of Modifications made by that Contributor with its
Contributor Version (or portions of such combination).
(c) The licenses granted
in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available to a third
party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted:
(1) for any code that Contributor has deleted from the Contributor Version; (2)
for infringements caused by: (i) third party modifications of Contributor
Version, or (ii) the combination of Modifications made by that Contributor with
other software (except as part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered Software in the absence of
Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or
otherwise make available in Executable form must also be made available in
Source Code form and that Source Code form must be distributed only under the
terms of this License. You must include a copy of this License with every copy
of the Source Code form of the Covered Software You distribute or otherwise make
available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a
reasonable manner on or through a medium customarily used for software
exchange.
3.2. Modifications.
The Modifications that You create or to which
You contribute are governed by the terms of this License. You represent that You
believe Your Modifications are Your original creation(s) and/or You have
sufficient rights to grant the rights conveyed by this License.
3.3. Required Notices.
You must include a notice in each of Your Modifications that
identifies You as the Contributor of the Modification. You may not remove or
alter any copyright, patent or trademark notices contained within the Covered
Software, or any notices of licensing or any descriptive text giving attribution
to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source
Code form that alters or restricts the applicable version of this License or the
recipients’ rights hereunder. You may choose to offer, and to charge a fee for,
warranty, support, indemnity or liability obligations to one or more recipients
of Covered Software. However, you may do so only on Your own behalf, and not on
behalf of the Initial Developer or any Contributor. You must make it absolutely
clear that any such warranty, support, indemnity or liability obligation is
offered by You alone, and You hereby agree to indemnify the Initial Developer
and every Contributor for any liability incurred by the Initial Developer or
such Contributor as a result of warranty, support, indemnity or liability terms
3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.
UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.
10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

The binary distribution of this product bundles these dependencies under the following license:
Protocol Buffer Java API 2.5.0

This license applies to all parts of Protocol Buffers except the following:

- Atomicops support for generic gcc, located in
  src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
  This file is copyrighted by Red Hat Inc.

- Atomicops support for AIX/POWER, located in
  src/google/protobuf/stubs/atomicops_internals_power.h.
  This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

For:
XML Commons External Components XML APIs 1.3.04

--------------------------------------------------------------------------------

By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:
- The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

The binary distribution of this product bundles these dependencies under the following license:
JUnit 4.11
Eclipse JDT Core 3.1.1

--------------------------------------------------------------------------------

(EPL v1.0)
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are not
derivative works of the Program.
“Contributor” means any person or entity that distributes the Program.

“Licensed Patents” mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to
reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such
derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other combinations
which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to
its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to
Recipient for claims brought by any other entity based on infringement of
intellectual property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby assumes sole
responsibility to secure any other intellectual property rights needed, if any.
For example, if a third party patent license is required to allow Recipient to
distribute the Program, it is Recipient's responsibility to acquire that license
before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright
rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively “Losses”) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order
to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor’s responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The binary distribution of this product bundles these dependencies under the following license:
JSch 0.1.54
ParaNamer Core 2.3
JLine 0.9.94
leveldbjni-all 1.8
Hamcrest Core 1.3
ASM Core 5.0.4
ASM Commons 5.0.2
ASM Tree 5.0.2

---------------------------------------------------------------------
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the organization nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The binary distribution of this product bundles these dependencies under the following license:

FindBugs-jsr305 3.0.0

dnsjava 2.1.7, Copyright (c) 1998-2011, Brian Wellington. All rights reserved.

(2-clause BSD)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

The binary distribution of this product bundles these dependencies under the following license:
"Java Concurrency in Practice" book annotations 1.0
--------------------------------------------------------------------------------
(CCAL v2.5)
THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on
the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work (“cover version”) and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of
the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., ”French translation of the Work by Original Author,” or ”Screenplay based on original Work by Original Author”). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

The binary distribution of this product bundles these dependencies under the following license:

```
jamon-runtime 2.4.1
```

(MPL 2.0)

Mozilla Public License

Version 2.0
1. Definitions

1.1. Contributor
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. Contributor Version
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributors Contribution.

1.3. Contribution
means Covered Software of a particular Contributor.

1.4. Covered Software
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. Incompatible With Secondary Licenses
means

that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. Executable Form
means any form of the work other than Source Code Form.

1.7. Larger Work
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. License
means this document.

1.9. Licensable
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. Modifications
means any of the following:

any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or
any new file in Source Code Form that contains any Covered Software.

1.11. Patent Claims of a Contributor
means any patent claim(s), including without limitation, method, process, and
apparatus claims, in any patent Licensable by such Contributor that would be
infringed, but for the grant of the License, by the making, using, selling,
offering for sale, having made, import, or transfer of either its Contributions
or its Contributor Version.

1.12. Secondary License
means either the GNU General Public License, Version 2.0, the GNU Lesser General
Public License, Version 2.1, the GNU Affero General Public License, Version 3.0,
or any later versions of those licenses.

1.13. Source Code Form
means the form of the work preferred for making modifications.

1.14. You (or Your)
means an individual or a legal entity exercising rights under this License. For
legal entities, You includes any entity that controls, is controlled by,
or is under common control with You. For purposes of this definition,
control means (a) the power, direct or indirect, to cause the direction or
management of such entity, whether by contract or otherwise, or (b) ownership of
more than fifty percent (50%) of the outstanding shares or beneficial ownership
of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive
license:

under intellectual property rights (other than patent or trademark) Licensable
by such Contributor to use, reproduce, make available, modify, display, perform,
distribute, and otherwise exploit its Contributions, either on an unmodified
basis, with Modifications, or as part of a Larger Work; and

under Patent Claims of such Contributor to make, use, sell, offer for sale, have
made, import, and otherwise transfer either its Contributions or its Contributor
Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become
effective for each Contribution on the date the Contributor first distributes
such Contribution.
2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- for any code that a Contributor has removed from Covered Software; or
- for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any
Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify
every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

6. Disclaimer of Warranty

Covered Software is provided under this License on an as is basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as
to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such partys negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a partys ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.
10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - Incompatible With Secondary Licenses Notice

This Source Code Form is Incompatible With Secondary Licenses, as defined by the Mozilla Public License, v. 2.0.

The binary distribution of this product bundles these dependencies under the following license:

JDOM 1.1

*****************************************************************************

/*...*/

Copyright (C) 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact <request_AT_jdom_DOT_org>.

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management <request_AT_jdom_DOT_org>.

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project (http://www.jdom.org)."

Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter <jhunter_AT_jdom_DOT_org> and Brett McLaughlin <brett_AT_jdom_DOT_org>. For more information on the JDOM Project, please see <http://www.jdom.org/>.

*/

The binary distribution of this product bundles these dependencies under the
This project bundles a derivative image for our Orca Logo. This image is available under the Creative Commons By Attribution 3.0 License.

Creative Commons Legal Code

Attribution 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and
arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital
performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to
collect royalties, whether individually or, in the event that the
Licensor is a member of a collecting society that administers
voluntary licensing schemes, via that society, from any exercise
by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now
known or hereafter devised. The above rights include the right to make
such modifications as are technically necessary to exercise the rights in
other media and formats. Subject to Section 8(f), all rights not expressly
granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made
subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms
of this License. You must include a copy of, or the Uniform Resource
Identifier (URI) for, this License with every copy of the Work You
Distribute or Publicly Perform. You may not offer or impose any terms
on the Work that restrict the terms of this License or the ability of
the recipient of the Work to exercise the rights granted to that
recipient under the terms of the License. You may not sublicense the
Work. You must keep intact all notices that refer to this License and
to the disclaimer of warranties with every copy of the Work You
Distribute or Publicly Perform. When You Distribute or Publicly
Perform the Work, You may not impose any effective technological
measures on the Work that restrict the ability of a recipient of the
Work from You to exercise the rights granted to that recipient under
the terms of the License. This Section 4(a) applies to the Work as
incorporated in a Collection, but this does not require the Collection
apart from the Work itself to be made subject to the terms of this
License. If You create a Collection, upon notice from any Licensor You
must, to the extent practicable, remove from the Collection any credit
as required by Section 4(b), as requested. If You create an
Adaptation, upon notice from any Licensor You must, to the extent
practicable, remove from the Adaptation any credit as required by
Section 4(b), as requested.

b. If You Distribute, or Publicly Perform the Work or any Adaptations or
Collections, You must, unless a request has been made pursuant to
Section 4(a), keep intact all copyright notices for the Work and
provide, reasonable to the medium or means You are utilizing: (i) the
name of the Original Author (or pseudonym, if applicable) if supplied,
and/or if the Original Author and/or Licensor designate another party
or parties (e.g., a sponsor institute, publishing entity, journal) for
attribution (“Attribution Parties”) in Licensor’s copyright notice,
terms of service or by other reasonable means, the name of such party
or parties; (ii) the title of the Work if supplied; (iii) to the
extent reasonably practicable, the URI, if any, that Licensor
specifies to be associated with the Work, unless such URI does not
refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4 (b) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

c. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR
7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention.
for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at https://creativecommons.org/.


Copyright (C) 2007 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of
the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2664 lucene-queries 7.4.0
1.2664.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was derived from unicode conversion examples available at
http://www.unicode.org/Public/PROGRAMS/CVTUTF. Here is the copyright from those sources:

/*
 * Copyright 2001-2004 Unicode, Inc.
 *
 * Disclaimer
 *
 * This source code is provided as is by Unicode, Inc. No claims are
 * made as to fitness for any particular purpose. No warranties of any
 * kind are expressed or implied. The recipient agrees to determine
 * applicability of information provided. If this file has been
 * purchased on magnetic or optical media from Unicode, Inc., the
 * sole remedy for any claim will be exchange of defective media
 * within 90 days of receipt.
 *
 * Limitations on Rights to Redistribute This Code
 *
 * Unicode, Inc. hereby grants the right to freely use the information
*/
* supplied in this file in the creation of products supporting the
* Unicode Standard, and to make copies of this file in any form
* for internal or external distribution as long as this notice
* remains attached.
*/

Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was
derived from Python 2.4.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from Python 3.1.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in core/src/java/org/apache/lucene/util/automaton was
derived from Brics automaton sources available at
www.brics.dk/automaton/. Here is the copyright from those sources:

/*
 * Copyright (c) 2001-2009 Anders Moeller
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 *    derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The levenshtein automata tables in core/src/java/org/apache/lucene/util/automaton were automatically generated with the moman/finenight FSA package.

Here is the copyright for those sources:

# Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
# Permission is hereby granted, free of charge, to any person
# obtaining a copy of this software and associated documentation
# files (the "Software"), to deal in the Software without
# restriction, including without limitation the rights to use,
# copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following
# conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
# OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
# FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
# OTHER DEALINGS IN THE SOFTWARE.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
 * Copyright (C) 1999-2010, International Business Machines
 * Corporation and others.  All Rights Reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, and/or sell copies of the
 * Software, and to permit persons to whom the Software is furnished to do so,
 * provided that the above copyright notice(s) and this permission notice appear
 * in all copies of the Software and that both the above copyright notice(s) and
 * this permission notice appear in supporting documentation.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
* IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
* LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
* ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
* IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* Except as contained in this notice, the name of a copyright holder shall not
* be used in advertising or otherwise to promote the sale, use or other
* dealings in this Software without prior written authorization of the
* copyright holder.
*/

The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  * this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  * notice, this list of conditions and the following disclaimer in the
  * documentation and/or other materials provided with the distribution.
* Neither the name of the copyright holders nor the names of its contributors
  * may be used to endorse or promote products derived from this software
  * without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names "Center for Intelligent Information Retrieval" and "University of Massachusetts" must not be used to endorse or promote products derived from this software without prior written permission. To obtain permission, contact info@ciir.cs.umass.edu.

THIS SOFTWARE IS PROVIDED BY UNIVERSITY OF MASSACHUSETTS AND OTHER CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Morfologik nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

The dictionary comes from Morfologik project. Morfologik uses data from Polish ispell/myspell dictionary hosted at http://www.sjp.pl/slownik/en/ and is licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike. The part-of-speech tags were added in Morfologik project and are not found in the data from sjp.pl. The tagset is similar to IPI PAN tagset.

---

The following license applies to the Morfeusz project, used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/

Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski, Marcin Woliski, Robert Woosz

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY COPYRIGHT HOLDERS AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects,
including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license
and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such
as the Unicode Character Database. See http://unicode.org/copyright.html for more
details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is
BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were
automatically generated with the moman/finenight FSA library, created by
Jean-Philippe Barrette-LaPierre. This library is available under an MIT license,
see http://sites.google.com/site/rrettesite/moman and
http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from
the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
The snowball stemmers in
    analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
    analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from
http://snowball.tartarus.org/

The KStem stemmer in
    analysis/common/src/org/apache/lucene/analysis/en
was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst)
under the BSD-license.

The Arabic, Persian, Romanian, Bulgarian, Hindi and Bengali analyzers (common) come with a default
stopword list that is BSD-licensed created by Jacques Savoy. These files reside in:
    analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt,
    analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt,
    analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt,
    analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt,
    analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt,
    analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt

The German, Spanish, Finnish, French, Hungarian, Italian, Portuguese, Russian and Swedish light stemmers
(common) are based on BSD-licensed reference implementations created by Jacques Savoy and
Ljiljana Dolamic. These files reside in:
    analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
    analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed
by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil,
and Edmond Nolan.

The Polish analyzer (stempel) comes with a default
stopword list that is BSD-licensed created by the Carrot2 project. The file resides
The SmartChineseAnalyzer source code (smartcn) was provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/) is derived from Unicode data such as the Unicode Character Database. See http://unicode.org/copyright.html for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary (http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP) (http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original source code for this can be found at http://www.eclipse.org/jetty/downloads.php

===========================================================================
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

===========================================================================
Mecab-ipadic-2.7.0-20070801 Notice
===========================================================================

Nara Institute of Science and Technology (NAIST), the copyright holders, disclaims all warranties with regard to this software, including all implied warranties of merchantability and fitness, in no event shall NAIST be liable for any special, indirect or consequential damages or any damages

See http://project.carrot2.org/license.html.
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
implied, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.

In the case where the program, whether in its original form or modified, was distributed or delivered to or received by a user from any person, organization or entity other than ICOT, unless it makes or grants independently of ICOT any specific warranty to the user in writing, such person, organization or entity, will also be exempted from and not be held liable to the user for any such damages as noted above as far as the program is concerned.

===========================================================================
Nori Korean Morphological Analyzer - Apache Lucene Integration
===========================================================================

This software includes a binary and/or source version of data from

mecab-ko-dic-2.0.3-20170922

which can be obtained from

https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz

1.2665 drill-jdbc-all 1.18.0
1.2665.1 Available under license :

JDBC JAR with all dependencies
Copyright 2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

apr 1.6.5

passport 0.4.0 0.4.0

Available under license:

The MIT License (MIT)

Copyright (c) 2011-2015 Jared Hanson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.2669 jline-terminal 3.12.1

1.2669.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2017, the original author or authors.
 * 
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
 * 
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-
jar/org/jline/utils/NonBlockingPumpInputStream.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-
jar/org/jline/terminal/impl/AbstractWindowsConsoleWriter.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-
jar/org/jline/utils/WriterOutputStream.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-
jar/org/jline/utils/NonBlockingPumpReader.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-
jar/org/jline/utils/PumpReader.java

No license file was found, but licenses were detected in source scan.

#
# Copyright (c) 2002-2016, the original author or authors.
#
# This software is distributable under the BSD license. See the terms of the
# BSD license in the documentation provided with this software.
#
# https://opensource.org/licenses/BSD-3-Clause
#
#
auto_left_margin, bw, bw
auto_right_margin, am, am
back_color_erase, bce, ut
can_change, ccc, cc
ceol_standout_glitch, xhp, xs
col_addr_glitch, xhpa, YA
cpi_changes_res, cpix, YF
cr_cancels_micro_mode, crxm, YB
dest_tabs_magic_smso, xt, xt
eat_newline_glitch, xenl, xn
erase_overstrike, eo, eo
generic_type, gn, gn
hard_copy, hc, hc
hard_cursor, chts, HC
has_meta_key, km, km
has_print_wheel, daisy, YC
has_status_line, hs, hs
hue_lightness_saturation, hls, hl
insert_null_glitch, in, in
lpi_changes_res, lpix, YG
memory_above, da, da
memory_below, db, db
move_insert_mode, mir, mi
move_standout_mode, msgr, ms
needs_xon_xoff, nxon, nx
no_esc_ctlc, xsb, xb
no_pad_char, npc, NP
non_dest_scroll_region, ndscr, ND
non_rev_rmcup, nrmc, NR
over_strike, os, os
prtr_silent, mc5i, 5i
row_addr_glitch, xvpa, YD
semi_auto_right_margin, sam, YE
status_line_esc_ok, eslok, es
tilde_glitch, hz, hz
transparent_underline, ul, ul
xon_xoff, xon, xo
columns, cols, co
init_tabs, it, it
label_height, lh, lh
label_width, lw, lw
lines, lines, li
lines_of_memory, lm, lm
magic_cookie_glitch, xmc, sg
max_attributes, ma, ma
max_colors, colors, Co
max_pairs, pairs, pa
maximum_windows, wnum, MW
no_color_video, ncv, NC
num_labels, nlab, Nl
padding_baud_rate, pb, pb
virtual_terminal, vt, vt
width_status_line, wsl, ws
delete_character, dch1, dc
delete_line, dl1, dl
dial_phone, dial, DI
dis_status_line, dsl, ds
display_clock, dclk, DK
down_half_line, hd, hd
en_a_cs, enacs, eA
enter_alt_charset_mode, smacs, as
enter_am_mode, smam, SA
enter_blink_mode, blink, mb
enter_bold_mode, bold, md
enter_ca_mode, smcup, ti
enter_delete_mode, smdc, dm
enter_dim_mode, dim, mh
enter_doublewide_mode, swidm, ZF
enter_draft_quality, sdrfq, ZG
enter_insert_mode, smir, im
enter_italics_mode, sitm, ZH
enter_leftward_mode, slm, ZI
enter_micro_mode, smicm, ZJ
enter_near_letter_quality, snlq, ZK
enter_normal_quality, snrmq, ZL
enter_protected_mode, prot, mp
enter_reverse_mode, rev, mr
enter_secure_mode, invis, mk
enter_shadow_mode, sshm, ZM
enter_standout_mode, smso, so
enter_subscript_mode, ssubm, ZN
enter_superscript_mode, ssupm, ZO
enter_underline_mode, smul, us
enter_upward_mode, sum, ZP
enter_xon_mode, smxon, SX
erase_chars, ech, ec
exit_alt_charset_mode, rmacs, ae
exit_am_mode, rmam, RA
exit_attribute_mode, sgr0, me
exit_ca_mode, rmcup, te
exit_delete_mode, rmdc, ed
exit_doublewide_mode, rwidm, ZQ
exit_insert_mode, rmir, ei
exit_italics_mode, ritm, ZR
exit_leftward_mode, rlm, ZS
exit_micro_mode, rmicm, ZT
exit_shadow_mode, rshm, ZU
exit_standout_mode, rmso, se
exit_subscript_mode, rsubm, ZV
exit_superscript_mode, rsupm, ZW
exit_underline_mode, rmul, ue
exit_upward_mode, rum, ZX
exit_xon_mode, rmxon, RX
fixed_pause, pause, PA
flash_hook, hook, fh
flash_screen, flash, vb
form_feed, ff, ff
from_status_line, fsl, fs
goto_window, wingo, WG
hangup, hup, HU
init_1string, is1, i1
init_2string, is2, is
init_3string, is3, i3
init_file, if, if
init_prog, iprog, iP
initialize_color, initc, Ic
initialize_pair, initp, Ip
insert_character, ich1, ic
insert_line, il1, al
insert_padding, ip, ip
key_a1, ka1, K1
key_a3, ka3, K3
key_b2, kb2, K2
key_backspace, kbs, kb
key_beg, kbeg, @1
key_btab, kcbt, kB
key_c1, kc1, K4
key_c3, kc3, K5
key_cancel, kcan, @2
key_catab, ktbc, ka
key_clear, kclr, kC
key_close, kclo, @3
key_command, kcmd, @4
key_copy, kcpy, @5
key_create, kcrt, @6
key_ctab, kctab, kt
key_dc, kdch1, kD
key_dl, kd11, kL
key_down, kcud1, kd
key_eic, krmir, kM
key_end, kend, @7
key_enter, kent, @8
key_eol, kel, kE
key_eos, ked, kS
key_exit, kext, @9
key_f0, kf0, k0
key_f1, kf1, k1
key_f10, kf10, k;
key_f56, kf56, Fk
key_f57, kf57, Fl
key_f58, kf58, Fm
key_f59, kf59, Fn
key_f6, kf6, k6
key_f60, kf60, Fo
key_f61, kf61, Fp
key_f62, kf62, Fq
key_f63, kf63, Fr
key_f7, kf7, k7
key_f8, kf8, k8
key_f9, kf9, k9
key_find, kfnd, @0
key_help, khlp, %1
key_home, khome, kh
key_ic, kich1, kl
key_il, kil1, kA
key_left, kcub1, kl
key_ll, kll, kH
key_mark, kmrk, %2
key_message, kmsg, %3
key_move, kmov, %4
key_next, knxt, %5
key_npage, knp, kN
key_open, kopn, %6
key_options, kopt, %7
key_ppage, kpp, kP
key_previous, kprv, %8
key_print, kprt, %9
key_redo, krdo, %0
key_reference, kref, &1
key_refresh, krfr, &2
key_replace, krpl, &3
key_restart, krst, &4
key_resume, kres, &5
key_right, kcuf1, kr
key_save, ksav, &6
key_sbeg, kBEG, &9
key_scancel, kCAN, &0
key_scommand, kCMD, *1
key_scopy, kCPY, *2
key_screate, kCRT, *3
key_sdc, kDC, *4
key sdl, kDL, *5
key_select, kslt, *6
key_send, kEND, *7
key_seol, kEOL, *8
key_sexit, kEXT, *9
key_sf, kind, kF
key_sfind, kFND, *0
key_shelp, kHLP, #1
key_shome, kHOM, #2
key_sic, kIC, #3
key_sleft, kLFT, #4
key_smessage, kMSG, %a
key_smov, kMOV, %b
key_snxt, kNXT, %c
key_sopt, kOPT, %d
key_sprevious, kPRV, %e
key_sprint, kPRT, %f
key_sr, kri, kR
key_sredo, kRDO, %g
key_srsplace, kRPL, %h
key_sright, kRIT, %i
key_srssume, kRES, %j
key_ssave, kSAV, !1
key_sssuspend, kSPD, !2
key_stab, khts, kT
key_sundo, kUND, !3
key_suspen, kspd, &7
key undo, kund, &8
key_up, kcuu1, ku
keypad_local, rmkx, ke
keypad_xmit, smkx, ks
lab_f0, lf0, l0
lab_f1, lf1, 11
lab_f10, lf10, la
lab_f2, lf2, 12
lab_f3, lf3, 13
lab_f4, lf4, 14
lab_f5, lf5, 15
lab_f6, lf6, 16
lab_f7, lf7, 17
lab_f8, lf8, 18
lab_f9, lf9, 19
label_format, fln, Lf
label_off, rmln, LF
label_on, smln, LO
meta_off, rmm, mo
meta_on, smm, mm
micro_column_address, mhpa, ZY
micro_down, mcud1, ZZ
micro_left, mcub1, Za
micro_right, mcuf1, Zb
micro_row_address, mvpa, Zc
micro_up, mcuu1, Zd
set_foreground, setf, Sf
set_left_margin, smgl, ML
set_left_margin_parm, smglp, Zm
set_right_margin, smgr, MR
set_right_margin_parm, smgrp, Zn
set_tab, hts, st
set_top_margin, smgt, Zo
set_top_margin_parm, smgtp, Zp
set_window, wind, wi
start_bit_image, sbim, Zq
start_char_set_def, scsd, Zr
stop_bit_image, rbim, Zs
stop_char_set_def, rcsd, Zt
subscript_characters, subcs, Zu
superscript_characters, supcs, Zv
tab, ht, ta
tone, tone, TO
underline_char, uc, uc
up_half_line, hu, hu
user0, u0, u0
user1, u1, u1
user2, u2, u2
user3, u3, u3
user4, u4, u4
user5, u5, u5
user6, u6, u6
user7, u7, u7
user8, u8, u8
user9, u9, u9
wait_tone, wait, WA
xon_character, xonc, XN
define_bit_image_region, defbi, Yx
device_type, devt, dv
display_pc_char, dispc, S1
define_bit_image_region, defbi, Yx
display_pc_charset_mode, smpch, S2
define_scancode_mode, smsc, S4
exit_pc_charset_mode, rmpch, S3
exit_scancode_mode, rmsc, S5
get_mouse, getm, Gm
key_mouse, kmous, Km
mouse_info, minfo, Mi
pc_term_options, pctrm, S6
pkey_plab, pxl, xl
req_mouse_pos, reqmp, RQ
scancode_escape, scesc, S7
set0_des_seq, s0ds, s0
set1_des_seq, s1ds, s1
set2_des_seq, s2ds, s2
set3_des_seq, s3ds, s3
set_a_background, setab, AB
set_a_foreground, setaf, AF
set_color_band, setcolor, Yz
set_lr_margin, smglr, ML
set_page_length, slines, YZ
set_tb_margin, smgtb, MT
enter_horizontal_hl_mode, ehhlm, Xh
enter_left_hl_mode, elhlm, Xl
enter_low_hl_mode, elohlm, Xo
enter_right_hl_mode, erhlm, Xr
enter_top_hl_mode, ethlm, Xt
enter_vertical_hl_mode, evhlm, Xv
set_a_attributes, sgr1, sA
set_pglen_inch, slength, sL

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/capabilities.txt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2016, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
 *
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/ExecPty.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/MouseSupport.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2018 the original author(s).
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-
  jar/org/jline/utils/AnsiWriter.java
No license file was found, but licenses were detected in source scan.

# # Copyright (c) 2002-2018, the original author or authors.
# # This software is distributable under the BSD license. See the terms of the
# # BSD license in the documentation provided with this software.
# # https://opensource.org/licenses/BSD-3-Clause
#
black
maroon
green
olive
navy
purple
teal
silver
grey
red
lime
yellow
blue
fuchsia
aqua
white
grey0
navyblue
darkblue
blue3
blue3a
blue1
darkgreen
deepskyblue4
deepskyblue4a
palegreen1
aquamarine1a
darkslategray1
red3
deeppink4b
mediumvioletred
magenta3
darkvioleta
purpleb
darkorange3
indianred
hotpink3
mediumorcid3
mediumorcid
mediumpurple2
darkgoldenrod
lightsalmon3
rosybrown
grey63
mediumpurple2a
mediumpurple1
gold3
darkkhaki
navajowhite3
grey69
lightsteelblue3
lightsteelblue
yellow3
darkolivegreen3b
darkseagreen3a
darkseagreen2
lightcyan3
lightskyblue1
greenyellow
darkolivegreen2
palegreen1a
darkseagreen2a
darkseagreen1
paleturquoise1
red3a
deeppink3
deeppink3a
magenta3a
magenta3b
magenta2
darkorange3a
indianreda
hotpink3a
hotpink2
orchid
mediumorchid1
orange3
lightsalmon3a
lightpink3
pink3
plum3
violet
gold3a
lightgoldenrod3
tan
mistyrose3
thistle3
plum2
yellow3a
khaki3
lightgoldenrod2
lightyellow3
grey84
lightsteelblue1
yellow2
darkolivegreen1
darkolivegreen1a
darkseagreen1a
honeydew2
lightcyan1
red1
deeppink2
deeppink1
deeppink1a
magenta2a
magenta1
orangered1
indianred1
indianred1a
hotpink
hotpinka
mediumorchid1a
darkorange
salmon1
lightcoral
palevioletred1
orchid2
orchid1
orange1
sandybrown
lightsalmon1
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/colors.txt
No license file was found, but licenses were detected in source scan.

2016, the original author or authors.

This software is distributable under the BSD license. See the terms of the
BSD license in the documentation provided with this software.

https://opensource.org/licenses/BSD

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/META-INF/maven/org.jline/jline-terminal/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2018, the original author or authors.
 * 
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
 * 
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/Colors.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/DumbTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/NonBlocking.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/Size.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/NonBlockingInputStream.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/PosixPtyTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/Curses.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/StyleResolver.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/AbstractTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/AttributedStyle.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/NonBlockingReaderImpl.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/PosixSysTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/terminal/Terminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar/org/jline/utils/Status.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/0/jline-terminal-3-12-1-sources-jar
jar/org/jline/utils/NonBlockingReader.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/LineDisciplineTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/Display.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/LineDisciplineTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/AttributedStringBuilder.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/NonBlockingInputStreamImpl.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/spi/JansiSupport.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/TerminalBuilder.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/AbstractWindowsTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/InfoCmp.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/AttributedCharSequence.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/spi/JnaSupport.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2019, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
 *
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/spi/fansiSupport.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/TerminalBuilder.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/AbstractWindowsTerminal.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/InfoCmp.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/utils/AttributedCharSequence.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/impl/AbstractPty.java
* /opt/ws_local/PERMITS_SQL/1057262552_1592255590.19/jline-terminal-3-12-1-sources-jar/org/jline/terminal/spi/JnaSupport.java

1.2670 regexpu-core 2.0.0
1.2670.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2671 onetime 2.0.1 2.0.1

1.2671.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2672 kafka-metadata 2.8.0

1.2672.1 Available under license :
Apache Kafka

Copyright 2021 The Apache Software Foundation.
This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at https://github.com/jersey/jersey/.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted by them:
Copyright (C) 2018 Lightbend Inc. <https://www.lightbend.com>
Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:
clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java
Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:
streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2673 0.3.6 0.3.6

1.2673.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2674 jsprim 1.4.1 1.4.1
1.2674.1 Available under license :
Copyright (c) 2012, Joyent, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE

1.2675 @babel/plugin-transform-for-of 7.16.0
1.2675.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANDABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2676 mimic-response 1.0.1
1.2676.1 Available under license:
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2677 should-util 1.0.1 1.0.1
1.2677.1 Available under license:
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1.2678 yargs 3.32.0 3.32.0

1.2678.1 Available under license:
Copyright 2010 James Halliday (mail@substack.net)

This project is free software released under the MIT/X11 license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2679 jcodings 1.0.55

1.2679.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Permission is hereby granted, free of charge, to any person obtaining a copy of
 * this software and associated documentation files (the "Software"), to deal in
 * the Software without restriction, including without limitation the rights to
 * use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
 * of the Software, and to permit persons to whom the Software is furnished to do
 * so, subject to the following conditions:
 * *
 * The above copyright notice and this permission notice shall be included in all
 */
* copies or substantial portions of the Software.
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/util/ArrayReader.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/ISO8859_5Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/specific/Universal_newline_Transcoder.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/specific/To_UTF_16BE_Transcoder.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/AbstractEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/CP949Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/GBKEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/UTF32BEEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/ISO8859_10Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/GBKEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Windows_31JEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Windows_1251Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/CP949Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/GbkEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/UTF32BEEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Windows_1251Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/CP949Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/BIg5Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/ascii/AsciiTables.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Windows_1254Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/specific/Iss2022jP_kddI_decoder_Transcoder.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/EConvResult.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/UTF16LEEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/BaseEUCJPEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/ISO8859_14Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/ISO8859_13Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/ISO8859_6Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/unicode/UnicodeCodeRange.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/specific/From_UTF_16BE_Transcoder.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/CaseFoldMapEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Gb2312Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/exception/ErrorMessage.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/transcode/specific/Cp50221_encoder_Transcoder.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Windows_1253Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/NonStrictEUCJPEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/EUCJPEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/Ko8Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/Config.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/unicode/UnicodeEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/ObjPtr.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/util/Hash.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/EncodingList.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/util/BytesHash.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-jar/org/jcodings/specific/BaseBIG5Encoding.java
jar/org/jcodings/specific/Big5UAOEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-
arj/org/jcodings/specific/NonStrictSJISEncoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-
jar/org/jcodings/specific/ISO8859_7Encoding.java
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-
jar/org/jcodings/transcode/Transcoder.java

No license file was found, but licenses were detected in source scan.

/
* Permission is hereby granted, free of charge, to any person obtaining a copy of
* this software and associated documentation files (the "Software"), to deal in
* the Software without restriction, including without limitation the rights to
* use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
* of the Software, and to permit persons to whom the Software is furnished to do
* so, subject to the following conditions:
* 
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
* 
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):
* /opt/cola/permits/1202447017_1631568351.65/0/jcodings-1-0-55-sources-
arj/org/jcodings/unicode/FixedWidthUnicodeEncoding.java

1.2680 url-parse 1.4.7 1.4.7

1.2680.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE Software.

1.2681 asm-util 7.3.1
1.2681.1 Available under license:

No license file was found, but licenses were detected in source scan.

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

Found in path(s):
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/CheckMethodAdapter.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/TraceFieldVisitor.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/Printer.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/CheckClassAdapter.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/CheckSignatureAdapter.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/CheckModuleAdapter.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/TraceModuleVisitor.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/TraceSignatureVisitor.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-jar/org/objectweb/asm/util/CheckAnnotationAdapter.java
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-
2011 INRIA, France Telecom
* All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-
  jar/org/objectweb/asm/util/package.html
No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework Copyright (c) 2000-2011 INRIA,
 * France Telecom All rights reserved.
 *<p>Redistribution and use in source and binary forms, with or without modification, are permitted
 * provided that the following conditions are met: 1. Redistributions of source code must retain the
 * above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions
 * in binary form must reproduce the above copyright notice, this list of conditions and the
 * following disclaimer in the documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its contributors may be used to
 * endorse or promote products derived from this software without specific prior written permission.
 *
 *<p>THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
 MERCHANTABILITY AND
 * FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
 OWNER OR
 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
 OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
 LIABILITY,
 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
 OTHERWISE) ARISING IN ANY
 * WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
 DAMAGE.
 */

Found in path(s):
- /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-
  jar/org/objectweb/asm/util/ASMifierSupport.java
- /opt/cola/permits/1143533479_1615689073.97/0/asm-util-7-3-1-sources-1-
  jar/org/objectweb/asm/util/TextifierSupport.java

1.2682 test-exclude 6.0.0
1.2682.1 Available under license:

Copyright (c) 2016, Contributors

Permission to use, copy, modify, and/or distribute this software
for any purpose with or without fee is hereby granted, provided
that the above copyright notice and this permission notice
appear in all copies.
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2683 get-package-type 0.1.0

1.2683.1 Available under license :
MIT License

Copyright (c) 2020 CFWare, LLC

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2684 pako 0.2.9

1.2684.1 Available under license :
(The MIT License)

Copyright (C) 2014-2016 by Vitaly Puzrin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2685 async 2.1.2 2.1.2
1.2685.1 Available under license:
Copyright (c) 2010-2016 Caolan McMahon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2686 babel-template 6.26.0 6.26.0
1.2686.1 Available under license:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2687 on-headers 1.0.2 1.0.2

1.2687.1 Available under license:
(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2688 pascalcase 0.1.1 0.1.1
1.2688.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2689 kind-of 4.0.0 4.0.0
1.2689.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
1.2690 undefsafe 2.0.5

1.2690.1 Available under license:

The MIT License (MIT)

Copyright 2016 Remy Sharp, http://remysharp.com <remy@remysharp.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2691 touch 3.1.0

1.2691.1 Available under license:

MIT License

Copyright (c) 2018 livelybone(2631541504@qq.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2692 strip-ansi 5.2.0 5.2.0
1.2692.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2693 jnr-posix 3.0.41
1.2693.1 Available under license:

Found license 'GNU Lesser General Public License' in "* License Version 1.0 (the "License"); you may not use this file * except in compliance with the License. You may obtain a copy of * Software distributed under the License is distributed on an "AS * Alternatively, the contents of this file may be used under the terms of * either of the GNU * General Public License Version 2 or later (the "GPL"), * or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), * use your version of this file under the terms of the CPL, indicate your'

1.2694 cli-cursor 1.0.2
1.2694.1 Available under license :

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2695 base64-js 0.0.2 0.0.2

1.2695.1 Available under license :

The MIT License (MIT)

Copyright (c) 2018 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.2696 isexe 2.0.0

1.2696.1 Available under license:

The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2697 d-conf 0.24.0-2

1.2697.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply, and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.
12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.
That's all there is to it!

1.2698 libx11 1.6.3-1ubuntu2.1

1.2698.1 Available under license:

The following is the 'standard copyright' agreed upon by most contributors, and is currently the canonical license preferred by the X.Org Foundation. This is a slight variant of the common MIT license form published by the Open Source Initiative at http://www.opensource.org/licenses/mit-license.php

Copyright holders of new code should use this license statement where possible, and insert their name to this list. Please sort by surname for people, and by the full name for other entities (e.g. Juliusz Chroboczek sorts before Intel Corporation sorts before Daniel Stone).

See each individual source file or directory for the license that applies to that file.

Copyright (C) 2003-2006,2008 Jamey Sharp, Josh Triplett
Copyright 2009 Red Hat, Inc.
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

------------------------------------------------------------------------
The following licenses are 'legacy' - usually MIT/X11 licenses with the name of the copyright holder(s) in the license statement:

Copyright 1984-1994, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

X Window System is a trademark of The Open Group.

----------------------------------------

Copyright 2000 The XFree86 Project, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.


Portions Copyright 1990, 1991 by Tektronix, Inc.

Permission to use, copy, modify and distribute this documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in all copies, and that the names of Digital and Tektronix not be used in in advertising or publicity pertaining to this documentation without specific, written prior permission. Digital and Tektronix makes no representations about the suitability of this documentation for any purpose. It is provided "as is" without express or implied warranty.

----------------------------------------

Copyright (c) 1999-2000 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE FREE SOFTWARE FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the Free Software Foundation shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the
Free Software Foundation.

Code and supporting documentation (c) Copyright 1990 1991 Tektronix, Inc. All Rights Reserved

This file is a component of an X Window System-specific implementation of Xcms based on the TekColor Color Management System. TekColor is a trademark of Tektronix, Inc. The term "TekHVC" designates a particular color space that is the subject of U.S. Patent No. 4,985,853 (equivalent foreign patents pending). Permission is hereby granted to use, copy, modify, sell, and otherwise distribute this software and its documentation for any purpose and without fee, provided that:

1. This copyright, permission, and disclaimer notice is reproduced in all copies of this software and any modification thereof and in supporting documentation;
2. Any color-handling application which displays TekHVC color coordinates identifies these as TekHVC color coordinates in any interface that displays these coordinates and in any associated documentation;
3. The term "TekHVC" is always used, and is only used, in association with the mathematical derivations of the TekHVC Color Space, including those provided in this file and any equivalent pathways and mathematical derivations, regardless of digital (e.g., floating point or integer) representation.

Tektronix makes no representation about the suitability of this software for any purpose. It is provided "as is" and with all faults.

TEKTRONIX DISCLAIMS ALL WARRANTIES APPLICABLE TO THIS SOFTWARE, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL TEKTRONIX BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA, OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE, OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR THE PERFORMANCE OF THIS SOFTWARE.

(c) Copyright 1995 FUJITSU LIMITED
This is source code modified by FUJITSU LIMITED under the Joint Development Agreement for the CDE/Motif PST.

Copiright 1992 by Oki Technosystems Laboratory, Inc.
Copyright 1992 by Fuji Xerox Co., Ltd.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Oki Technosystems Laboratory and Fuji Xerox not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.
Oki Technosystems Laboratory and Fuji Xerox make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OKI TECHNOSYSTEMS LABORATORY AND FUJI XEROX BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

-------------------------------


Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.
FUJITSU LIMITED makes no representations about the suitability of this software for any purpose.
It is provided "as is" without express or implied warranty.

FUJITSU LIMITED DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

-------------------------------
Copyright (c) 1995 David E. Wexelblat. All rights reserved

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DAVID E. WEXELBLAT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of David E. Wexelblat shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from David E. Wexelblat.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name OMRON not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTUOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of Open Software Foundation
not be used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission. Open Software
Foundation makes no representations about the suitability of this
software for any purpose. It is provided "as is" without express or
implied warranty.

OPEN SOFTWARE FOUNDATION DISCLAIMS ALL WARRANTIES WITH REGARD TO
THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL OPEN SOFTWARE FOUNDATION BE
LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright 1993, 1994                  by Sony Corporation

Permission to use, copy, modify, distribute, and sell this software and
its documentation for any purpose is hereby granted without fee, provided
that the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of FUJITSU LIMITED and Sony Corporation
not be used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission. FUJITSU LIMITED and
Sony Corporation makes no representations about the suitability of this
software for any purpose. It is provided "as is" without express or
implied warranty.

FUJITSU LIMITED AND SONY CORPORATION DISCLAIMS ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL FUJITSU LIMITED OR SONY CORPORATION BE LIABLE
FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE
USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright (c) 1993, 1995 by Silicon Graphics Computer Systems, Inc.

Permission to use, copy, modify, and distribute this
software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Silicon Graphics not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission. Silicon Graphics makes no representation about the suitability of this software for any purpose. It is provided "as is" without any express or implied warranty.

SILICON GRAPHICS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1993 by Digital Equipment Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FUJITSU LIMITED and Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FUJITSU LIMITED and Digital Equipment Corporation makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

FUJITSU LIMITED AND DIGITAL EQUIPMENT CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU LIMITED AND DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright 1993 by SunSoft, Inc.
Copyright 1999-2000 by Bruno Haible

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of SunSoft, Inc. and Bruno Haible not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. SunSoft, Inc. and Bruno Haible make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

SunSoft Inc. AND Bruno Haible DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL SunSoft, Inc. OR Bruno Haible BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the TOSHIBA Corp.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Open Software Foundation and TOSHIBA not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Open Software Foundation and TOSHIBA make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OPEN SOFTWARE FOUNDATION AND TOSHIBA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OPEN SOFTWARE FOUNDATION OR TOSHIBA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1988 by Wyse Technology, Inc., San Jose, Ca.,

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name Wyse not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

WYSE DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1991 by the Open Software Foundation
Copyright 1993, 1994 by the Sony Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Open Software Foundation and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Open Software Foundation and Sony Corporation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OPEN SOFTWARE FOUNDATION AND SONY CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OPEN SOFTWARE FOUNDATION OR SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1992, 1993 by FUJITSU LIMITED
Copyright 1993 by Fujitsu Open Systems Solutions, Inc.

Permission to use, copy, modify, distribute and sell this software
and its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appear in all copies and
that both that copyright notice and this permission notice appear
in supporting documentation, and that the name of FUJITSU LIMITED and
Fujitsu Open Systems Solutions, Inc. not be used in advertising or
publicity pertaining to distribution of the software without specific,
written prior permission.
FUJITSU LIMITED and Fujitsu Open Systems Solutions, Inc. makes no
representations about the suitability of this software for any purpose.
It is provided "as is" without express or implied warranty.

FUJITSU LIMITED AND FUJITSU OPEN SYSTEMS SOLUTIONS, INC. DISCLAIMS ALL
WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJITSU OPEN SYSTEMS
SOLUTIONS, INC. AND FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT
OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE
OF THIS SOFTWARE.

----------------------------------------

Copyright 1993, 1994 by Sony Corporation

Permission to use, copy, modify, distribute, and sell this software
and its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appear in all copies and
that both that copyright notice and this permission notice appear
in supporting documentation, and that the name of Sony Corporation
not be used in advertising or publicity pertaining to distribution
of the software without specific, written prior permission.
Sony Corporation makes no representations about the suitability of
this software for any purpose. It is provided "as is" without
express or implied warranty.

SONY CORPORATION DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO
EVENT SHALL SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF

Open Source Used In DNAC Platform Arica 12905
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1986, 1998 The Open Group
Copyright (c) 2000 The XFree86 Project, Inc.

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation.

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
X CONSORTIUM OR THE XFREE86 PROJECT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Except as contained in this notice, the name of the X Consortium or of the
XFree86 Project shall not be used in advertising or otherwise to promote the
sale, use or other dealings in this Software without prior written
authorization from the X Consortium and the XFree86 Project.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation,
and Nippon Telegraph and Telephone Corporation
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the names of OMRON, NTT Software, NTT, and
Open Software Foundation not be used in advertising or publicity
pertaining to distribution of the software without specific,
written prior permission. OMRON, NTT Software, NTT, and Open Software
Foundation make no representations about the suitability of this
software for any purpose. It is provided "as is" without express or
implied warranty.

OMRON, NTT SOFTWARE, NTT, AND OPEN SOFTWARE FOUNDATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON, NTT SOFTWARE, NTT, OR OPEN SOFTWARE FOUNDATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1988 by Wyse Technology, Inc., San Jose, Ca,
Copyright 1987 by Digital Equipment Corporation, Maynard, Massachusetts,

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name Digital not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

DIGITAL AND WYSE DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DIGITAL OR WYSE BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1991, 1992 by Fuji Xerox Co., Ltd.
Copyright 1992, 1993, 1994 by FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Fuji Xerox, FUJITSU LIMITED not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Fuji Xerox, FUJITSU LIMITED make no representations
about the suitability of this software for any purpose.
It is provided "as is" without express or implied warranty.

FUJI XEROX, FUJITSU LIMITED DISCLAIM ALL WARRANTIES WITH
REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL FUJI XEROX,
FUJITSU LIMITED BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA
OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 2006 Josh Triplett

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
 ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

----------------------------------------

(c) Copyright 1996 by Sebastien Marineau and Holger Veit
<marineau@genie.uottawa.ca>
    <Holger.Veit@gmd.de>

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL HOLGER VEIT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Sebastien Marineau or Holger Veit shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from Holger Veit or Sebastien Marineau.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation, and Nippon Telegraph and Telephone Corporation
Copyright 1991 by the Open Software Foundation
Copyright 1993 by the TOSHIBA Corp.
Copyright 1993, 1994 by Sony Corporation
Copyright 1993, 1994 by the FUJITSU LIMITED

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of OMRON, NTT Software, NTT, Open Software Foundation, and Sony Corporation not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON, NTT Software, NTT, Open Software Foundation, and Sony Corporation make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON, NTT SOFTWARE, NTT, OPEN SOFTWARE FOUNDATION, AND SONY CORPORATION DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON, NTT SOFTWARE, NTT, OPEN SOFTWARE FOUNDATION, OR SONY CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------
Copyright 2000 by Bruno Haible

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Bruno Haible not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Bruno Haible makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Bruno Haible DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL Bruno Haible BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 2003 Keith Packard

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Keith Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Keith Packard makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

KEITH PACKARD DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL KEITH PACKARD BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright (c) 2007-2009, Troy D. Hanson
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----------------------------------------

Copyright 1992, 1993 by TOSHIBA Corp.

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided
that the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of TOSHIBA not be used in advertising
or publicity pertaining to distribution of the software without specific,
written prior permission. TOSHIBA make no representations about the
suitability of this software for any purpose. It is provided "as is"
without express or implied warranty.

TOSHIBA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL
TOSHIBA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR
ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.

----------------------------------------

Copyright IBM Corporation 1993

All Rights Reserved

License to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of IBM not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

IBM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS, IN NO EVENT SHALL IBM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

----------------------------------------

Copyright 1990, 1991 by OMRON Corporation, NTT Software Corporation, and Nippon Telegraph and Telephone Corporation

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of OMRON, NTT Software, and NTT not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. OMRON, NTT Software, and NTT make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OMRON, NTT SOFTWARE, AND NTT, DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL OMRON, NTT SOFTWARE, OR NTT, BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

This notice applies to the files in this directory. They are taken from the libiconv-1.1 package, which is covered by the LGPL license. The files in this directory have been placed under the following copyright, with permission from the Free Software Foundation.

Copyright (c) 1999-2000 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE FREE SOFTWARE FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the Free Software Foundation shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the Free Software Foundation.

Notes:

1. This copyright applies only to the files in this directory, and not to the remaining files in libiconv.

2. The Free Software Foundation does not encourage the use of the above license for newly written software.

1.2699 log4j-api 2.14.0

1.2699.1 Available under license:

Apache Log4j API
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.2700 revalidator 0.1.8 0.1.8

1.2700.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright (c) 2009-2010 Alexis Sellier, Charlie Robbins, Nodejitsu Inc.

1.2701 caffeine 2.7.0
1.2702 esquery 1.3.1
1.2702.1 Available under license:

Copyright (c) 2013, Joel Feenstra
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the ESQuery nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JOEL FEENSTRA BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2703 regjsparser 0.7.0
1.2703.1 Available under license:

Copyright (c) Julian Viereck and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2704 commons-beanutils-core 1.8.0

1.2704.1 Available under license:

Apache Commons BeanUtils
Copyright 2000-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licenser" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, 
reproduction, and distribution of the Work otherwise complies with 
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, 
any Contribution intentionally submitted for inclusion in the Work 
by You to the Licensor shall be under the terms and conditions of 
this License, without any additional terms or conditions. 
Notwithstanding the above, nothing herein shall supersede or modify 
the terms of any separate license agreement you may have executed 
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade 
names, trademarks, service marks, or product names of the Licensor, 
except as required for reasonable and customary use in describing the 
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or 
agreed to in writing, Licensor provides the Work (and each 
Contributor provides its Contributions) on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or 
 implied, including, without limitation, any warranties or conditions 
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A 
PARTICULAR PURPOSE. You are solely responsible for determining the 
appropriateness of using or redistributing the Work and assume any 
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, 
whether in tort (including negligence), contract, or otherwise, 
unless required by applicable law (such as deliberate and grossly 
negligent acts) or agreed to in writing, shall any Contributor be 
liable to You for damages, including any direct, indirect, special, 
incidental, or consequential damages of any character arising as a 
result of this License or out of the use or inability to use the 
Work (including but not limited to damages for loss of goodwill, 
work stoppage, computer failure or malfunction, or any and all 
other commercial damages or losses), even if such Contributor 
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing 
the Work or Derivative Works thereof, You may choose to offer, 
and charge a fee for, acceptance of support, warranty, indemnity, 
or other liability obligations and/or rights consistent with this 
License. However, in accepting such obligations, You may act only 
on Your own behalf and on Your sole responsibility, not on behalf 
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2705 p-locate 4.1.0 4.1.0

1.2705.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
1.2706 tinycolor2 1.4.1 1.4.1

1.2706.1 Available under license:

Copyright (c), Brian Grinstead, http://briangrinstead.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2707 esprima 1.2.2 1.2.2

1.2707.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**1.2708 is-string 1.0.5**

1.2708.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.2709 chai 4.3.4**

1.2709.1 Available under license :

MIT License

Copyright (c) 2017 Chai.js Assertion Library

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2710 jetty-security 9.3.27.v20190418

1.2711 decode-uri-component 0.2.0 0.2.0

1.2711.1 Available under license:

The MIT License (MIT)

Copyright (c) Sam Verschueren <sam.verschueren@gmail.com> (github.com/SamVerschueren)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2712 grpc-protobuf 1.27.1

1.2712.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2713 jsreport-core 2.10.1 2.10.1

1.2713.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2714 component-inherit 0.0.3 0.0.3

1.2714.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2715 concat-stream 1.6.2 1.6.2

1.2715.1 Available under license:

The MIT License

Copyright (c) 2013 Max Ogden

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2716 braces 3.0.2 3.0.2
1.2716.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2018, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2717 p-locate 3.0.0 3.0.0
1.2717.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2718 chokidar 1.7.0

1.2718.1 Available under license:

No license file was found, but licenses were detected in source scan.

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Found in path(s):

* /opt/cola/permits/1127119904_1611787887.02/0/chokidar-1-7-0-2-tgz/package/README.md

1.2719 accepts 1.2.13 1.2.13

1.2719.1 Available under license:

(The MIT License)

Copyright (c) 2014 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2720 readable-stream 2.3.7 2.3.7

1.2720.1 Available under license:

Node.js is licensed for use as follows:

```---
Copyright Node.js contributors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.
---
```

This license applies to parts of Node.js originating from the
https://github.com/joyent/node repository:

```---
Copyright Joyent, Inc. and other Node contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
```
1.2721 rechoir 0.6.2 0.6.2

1.2721.1 Available under license:
Copyright (c) 2015 Tyler Kellen

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

1.2722 jgrapht-core 1.3.0

1.2722.1 Available under license:
Copyright (c) 2001-2014, JGraph Ltd
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the JGraph nor the
  names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JGRAPH BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Contributors ##

Copyright Notice: The JGraphT project source code is a composite of contributions by multiple authors. The copyright for each contribution is owned by the corresponding author. For details, please see [HISTORY.md](HISTORY.md) as well as the complete git history. All authors have agreed to license their contributions to the public as open source under the specific terms noted in [README.md](README.md).

JGraphT wouldn't be the library it is today without the source contributions and suggestions made by the authors:

- [Barak Naveh](http://sourceforge.net/users/barak_naveh/) (project founder)
- [John V Sichi](https://github.com/jsichi) (current project administrator)
- [Joris Kinable](https://github.com/jkinable) (JGraphT Project Reviewer/Committer and Release Manager)
- [Dimitrios Michail](https://github.com/d-michail) (JGraphT Project Reviewer/Committer)
- [Liviu Rau](http://sourceforge.net/users/liviu_aurelian/)
- [Nathan Fiedler](http://www.bluemarsh.com/personal/index.html)
- [Michael Behrisch](http://sourceforge.net/users/behrisch/)
- [Linda Buisman](http://sourceforge.net/users/linda_buisman/)
- Erik Postma
- Mikael Hansen
- Avner Linder
- Marden Neubert
- [Christian Soltenborn](http://sourceforge.net/users/csoltenborn/)
- [Christian Hammer](http://sourceforge.net/users/hammerc/)
- Ewgenij Proschak
- [Hartmut Benz](http://sourceforge.net/users/ivins/)
- [Charles Fry](http://frogcircus.org/)
- Guillaume Boulmier
- Carl Anderson
- Khanh Vu
- Aaron Harnly
- Welson Sun
- Trevor Harmon
- David Black-Schaffer
- Vinayak Borkar
- Andrew Berman
- Lucas Scharenbroich
- Hookahey
- Rita Dobler
- Szabolcs Besenyei
- Luiz Kill
- Christophe Thiebaud
- Jon Robinson
- Thomas Breitbart
- Sarah Komla-Ebri
- Graeme Ahokas
- Christoph Zauner
- [Andrew Gainer-Dewar](https://github.com/agdphd)
- Benedikt Waldvogel
- Victor Mikhailov
- Nils Olberg
- [Daniel Gomez-Sanchez](https://github.com/magicDGS)
- [Skuratovich Sergey](https://github.com/SSNikolaevich)
- [Martin Sturm](https://github.com/WorstCase00)
- [Patrick Sharp](https://github.com/sharpTrick)
- [Piotr Turski](https://github.com/piotrturski)
- [Alexandru Vleanu](https://github.com/AlexandruValeanu)
- [Davide Cavestro](https://github.com/davidecavestro)
- [Mark Raynsford](https://github.com/io7m)
- [Mariusz Smykua](https://github.com/mariuszs)
- [Pratik Tibrewal](https://github.com/tibrewalpratik17)
- [Chen Kui](https://github.com/Yimisim)
- [Timofey Chudakov](https://github.com/Toptachamann)
- [Konstantinos Karatsenidis](https://github.com/gate2k1)
- [Kirill Vishnyakov](https://github.com/LightnessOfBeing)
- [Emilio Cruciani](https://github.com/ioemilio)
- [Vivek Talreja](https://github.com/Vivek1012)
- [Gilles Gosuin](https://github.com/gilles-gosuin)
- [Viktor Volkov](https://github.com/chupacabra007)
- [Philipp Kaesgen](https://github.com/PhilippKaesgen)
- [Lukas Harzenetter](https://github.com/lharzenetter)
- [Christoph Grne](https://github.com/christophgruene)
- [Daniel Mock](https://github.com/danielmock)
- [Oliver Feith](https://github.com/Watercrystal)
- [Abdallah Atouani](https://github.com/AbdallahAt)
- [Peter Harman](https://github.com/harmanpa)
- [Nikhil Sharma](https://github.com/nks1558)
- [Dennis Fischer](https://github.com/pdelvo)
- [PHaroZ](https://github.com/PHaroZ)

(if we have missed your name on this list, please email us to get it fixed).

Other people have also helped in different ways: reporting bugs, requesting features, commenting, and by merely asking very good questions.

Many thanks to all of you.
GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.
In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from
such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution "originates” from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor” means any person or entity that Distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall
apply to the combination of the Contribution and the Program if, at the
time the Contribution is added by the Contributor, such addition
of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other
combinations which include the Contribution. No hardware per se is
licensed hereunder.

c) Recipient understands that although each Contributor grants the
licenses to its Contributions set forth herein, no assurances are
provided by any Contributor that the Program does not infringe the
patent or other intellectual property rights of any other entity.
Each Contributor disclaims any liability to Recipient for claims
brought by any other entity based on infringement of intellectual
property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby
assumes sole responsibility to secure any other intellectual
property rights needed, if any. For example, if a third party
patent license is required to allow Recipient to Distribute the
Program, it is Recipient's responsibility to acquire that license
before distributing the Program.

d) Each Contributor represents that to its knowledge it has
sufficient copyright rights in its Contribution, if any, to grant
the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no
Contributor makes additional grants to any Recipient (other than
those set forth in this Agreement) as a result of such Recipient's
receipt of the Program under the terms of a Secondary License
(if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in
accordance with section 3.2, and the Contributor must accompany
the Program with a statement that the Source Code for the Program
is available under this Agreement, and informs Recipients how to
obtain it in a reasonable manner on or through a medium customarily
used for software exchange; and

b) the Contributor may Distribute the Program under a license
different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all
      warranties and conditions, express and implied, including
      warranties or conditions of title and non-infringement, and
      implied warranties or conditions of merchantability and fitness
for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),
Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

[The "BSD 3-clause license"]
Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

MIT License for codepointat.js from https://git.io/codepointat
MIT License for fromcodepoint.js from https://git.io/vDW1m

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE 
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION 
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION 
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, 
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by 
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all 
other entities that control, are controlled by, or are under common 
control with that entity. For the purposes of this definition, 
"control" means (i) the power, direct or indirect, to cause the 
direction or management of such entity, whether by contract or 
otherwise, or (ii) ownership of fifty percent (50%) or more of the 
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity 
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, 
including but not limited to software source code, documentation 
source, and configuration files.

"Object" form shall mean any form resulting from mechanical 
transformation or translation of a Source form, including but 
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2723 superagent 3.8.3 3.8.3

1.2723.1 Available under license :
(The MIT License)

Copyright (c) 2014-2016 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2724 apache-log4j 1.2.17

1.2724.1 Available under license:

Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
1.2726 drill-shaded-guava 28.2-jre

1.2727 antlr-runtime 4.9

1.2727.1 Available under license:

No license file was found, but licenses were detected in source scan.

~ Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Found in path(s):
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/META-INF/maven/org.antlr/antlr4-runtime/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.
 * Use of this file is governed by the BSD 3-clause license that
 * can be found in the LICENSE.txt file in the project root.
 */

Found in path(s):
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/Lexer.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/misc/IntegerStack.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/ANTLRInputStream.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/tree/xpath/XPathRuleAnywhereElement.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/tree/ParseTreeWalker.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/UnbufferedCharStream.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/tree/ParseTreeWalker.java
* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-jar/org/antlr/v4/runtime/tree/TerminalNodeImpl.java
Open Source Used In DNAC Platform Arica 12976
<table>
<thead>
<tr>
<th>Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jar/org/antlr/v4/runtime/ATNDeserializer.java</td>
<td>/opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/BailErrorStrategy.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/EmptyPredictionContext.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/ATNDeserializer.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/AmbiguityInfo.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/tree/pattern/TokenTagToken.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/RecognitionException.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/LexerNoViableAltException.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/StarLoopbackState.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/IntStream.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/CharStreams.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/lexer/LexerActionExecutor.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/AbstractPredicateTransition.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/token/ToIntStream.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/lexer/FailedPredicateException.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/tree/pattern/TextChunk.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/tree/pattern/ParseTreeMatch.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/token/ConsoleErrorListener.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/SemanticContext.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/tree/xpath/XPathRuleElement.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/ArrayPredictionContext.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/atn/RangeTransition.java</td>
</tr>
<tr>
<td>* /opt/cola/permits/1258286449_1643052522.45/0/antlr4-runtime-4-9-sources-</td>
<td>jar/org/antlr/v4/runtime/tree/xpath/XPathLexerErrorListener.java</td>
</tr>
</tbody>
</table>
1.2728 archiver-utils 2.1.0 2.1.0

1.2728.1 Available under license:

Copyright (c) 2015 Chris Talkington.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
1.2731.1 Available under license:
(The MIT License)

Copyright (c) 2014 Guillermo Rauch <guillermo@learnboost.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2732 libxext 1.3.3-1
1.2732.1 Available under license:


Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be
used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

Copyright (c) 1996 Digital Equipment Corporation, Maynard, Massachusetts.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY CLAIM, DAMAGES, INCLUDING, BUT NOT LIMITED TO CONSEQUENTIAL OR INCIDENTAL DAMAGES, OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Digital Equipment Corporation shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from Digital Equipment Corporation.

Copyright (c) 1997 by Silicon Graphics Computer Systems, Inc.
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Silicon Graphics not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission. Silicon Graphics makes no representation about the suitability of this software for any purpose. It is provided "as is" without any express or implied warranty.

SILICON GRAPHICS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Copyright 1992 Network Computing Devices

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of NCD. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. NCD. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

NCD. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NCD. BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.


All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of Digital or Olivetti not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

DIGITAL AND OLIVETTI DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THEY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 1986, 1987, 1988 by Hewlett-Packard Corporation

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Hewlett-Packard not be used in
advertising or publicity pertaining to distribution of the software without specific, written prior permission.

Hewlett-Packard makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Copyright (c) 1994, 1995 Hewlett-Packard Company

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL HEWLETT-PACKARD COMPANY BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the Hewlett-Packard Company shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the Hewlett-Packard Company.

Copyright Digital Equipment Corporation, 1996

Permission to use, copy, modify, distribute, and sell this documentation for any purpose is hereby granted without fee, provided that the above copyright notice and this permission notice appear in all copies. Digital Equipment Corporation makes no representations about the suitability for any purpose of the information in this document. This documentation is provided "as is" without express or implied warranty.

Copyright (c) 1999, 2005, 2006, 2013, Oracle and/or its affiliates.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next paragraph) shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 1989 X Consortium, Inc. and Digital Equipment Corporation.
Copyright (c) 1992 X Consortium, Inc. and Intergraph Corporation.
Copyright (c) 1993 X Consortium, Inc. and Silicon Graphics, Inc.
Copyright (c) 1994, 1995 X Consortium, Inc. and Hewlett-Packard Company.

Permission to use, copy, modify, and distribute this documentation for any purpose and without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. Digital Equipment Corporation, Intergraph Corporation, Silicon Graphics, Hewlett-Packard, and the X Consortium make no representations about the suitability for any purpose of the information in this document. This documentation is provided "as is" without express or implied warranty.

1.2733 mongodb 3.6.6 3.6.6
1.2733.1 Available under license:
MIT License

Copyright (c) Microsoft Corporation.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2734 bluebird 3.4.7 3.4.7

1.2734.1 Available under license:

The MIT License (MIT)

Copyright (c) 2013-2015 Petka Antonov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2735 jersey-container-jetty-servlet 2.25.1

1.2735.1 Available under license:

Found license 'General Public License 2.0' in "Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License"
1.2736 debug 4.2.0 4.2.0
1.2736.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2737 electron-to-chromium 1.3.55 1.3.55
1.2737.1 Available under license:
Copyright 2018 Kilian Valkhof

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
1.2738 @babel/plugin-transform-computed-properties 7.16.0

1.2738.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2739 pcre 8.38-3.1

1.2739.1 Available under license:
PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End
PCRE LICENCE

PCRE is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as
specified below. The documentation for PCRE, supplied in the "doc"
directory, is distributed under the same terms as the software itself. The data
in the testdata directory is not copyrighted and is in the public domain.
The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions, and a just-in-time compiler that can be used to optimize pattern matching. These are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS
----------------------------
Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2015 University of Cambridge
All rights reserved.

PCRE JUST-IN-TIME COMPILATION SUPPORT
---------------------------------------
Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2010-2015 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER
--------------------------------
Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2009-2015 Zoltan Herczeg
All rights reserved.

THE C++ WRAPPER FUNCTIONS
---------------------------
Contributed by: Google Inc.
THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2740 kotlin 1.3.72
1.2740.1 Available under license :
No license file was found, but licenses were detected in source scan.
{"version":3,"file":"kotlin.js","sources":["wrapper.js","arrayUtils.js","callableReferenceUtils.js","conversions.js","c
ore.js","long.js","markerFunctions.js","misc.js","polyfills.js","rtti.js","arrayUtils.kt","runtime/Enum.kt","primitiveC
ompanionObjects.kt","generated/_Arrays.kt","generated/_Ranges.kt","kotlin/UByte.kt","kotlin/UInt.kt","kotlin/USh
ort.kt","Ranges.kt","collections/Collections.kt","collections/Maps.kt","collections/Sets.kt","text/StringNumberConv
ersions.kt","kotlin/UnsignedUtils.kt","collections/Iterables.kt","collections/Sequences.kt","util/Preconditions.kt","ge
nerated/_ArraysJs.kt","comparisons/Comparisons.kt","kotlin/Comparator.kt","util/Standard.kt","generated/_Compar
isonsJs.kt","generated/_Collections.kt","kotlin/collections.kt","collections/Iterators.kt","generated/_Comparisons.kt"
,"generated/_Maps.kt","generated/_Sequences.kt","generated/_Sets.kt","generated/_Strings.kt","text/Strings.kt","kot
lin/text/string.kt","kotlin/UByteArray.kt","kotlin/UIntArray.kt","kotlin/ULongArray.kt","kotlin/ULong.kt","kotlin/U
ShortArray.kt","generated/_UArrays.kt","generated/_UCollections.kt","generated/_UComparisons.kt","generated/_
URanges.kt","generated/_USequences.kt","Collections.kt","Iterators.kt","ProgressionIterators.kt","Progressions.kt",
"Range.kt","Unit.kt","annotation/Annotations.kt","internal/InternalAnnotations.kt","internal/progressionUtil.kt","ref
lect/KParameter.kt","reflect/KType.kt","reflect/KVariance.kt","reflect/KVisibility.kt","kotlin/builtins.kt","kotlin/jsT
ypeOf.kt","kotlin/kotlin.kt","kotlin/coroutines/CoroutineImpl.kt","util/Result.kt","coroutines/Continuation.kt","kotli
n/coroutines/intrinsics/IntrinsicsJs.kt","kotlin/currentBeMisc.kt","kotlin/jsOperators.kt","kotlin/math_jsv1.kt","kotlin/numbers_js-v1.kt","kotlin/text/numberConversions_jsv1.kt","generated/_CollectionsJs.kt","generated/_StringsJs.kt","generated/_UArraysJs.kt","jquery/ui.kt","kotlin/anno
tations.kt","kotlin/annotationsJVM.kt","kotlin/collections/AbstractMutableCollection.kt","kotlin/collections/Abstrac
tMutableList.kt","kotlin/collections/AbstractMutableMap.kt","kotlin/collections/AbstractMutableSet.kt","kotlin/coll
ections/ArrayList.kt","kotlin/collections/ArraySorting.kt","kotlin/collections/ArraysJs.kt","kotlin/collections/Equalit
yComparator.kt","kotlin/collections/HashMap.kt","kotlin/collections/HashSet.kt","kotlin/collections/InternalHashCo
deMap.kt","kotlin/collections/InternalMap.kt","kotlin/collections/InternalStringMap.kt","kotlin/collections/LinkedH
ashMap.kt","kotlin/collections/LinkedHashSet.kt","kotlin/concurrent.kt","kotlin/console.kt","kotlin/coroutines/Safe
ContinuationJs.kt","kotlin/coroutines/js/internal/EmptyContinuation.kt","kotlin/date.kt","kotlin/dom/Builders.kt","k
otlin/dom/Classes.kt","text/regex/RegexExtensions.kt","text/StringBuilder.kt","kotlin/dom/Dom.kt","kotlin/dom/Ev
entListener.kt","kotlin/dom/ItemArrayLike.kt","kotlin/dom/Mutations.kt","kotlin/dynamic.kt","kotlin/exceptionUtils
.kt","kotlin/exceptions.kt","kotlin/grouping.kt","collections/Grouping.kt","kotlin/js.math.kt","kotlin/json.kt","kotlin/
math.kt","kotlin/numbers.kt","kotlin/promise.kt","kotlin/random/PlatformRandom.kt","kotlin/reflect/JsClass.kt","ko
tlin/reflect/KClassImpl.kt","kotlin/reflect/KTypeHelpers.kt","kotlin/reflect/KTypeImpl.kt","kotlin/reflect/KTypePar
ameterImpl.kt","kotlin/reflect/primitives.kt","kotlin/reflect/reflection.kt","kotlin/regexp.kt","kotlin/sequence.kt","kot
lin/text/CharacterCodingExceptionJs.kt","kotlin/text/StringBuilderJs.kt","kotlin/text/char.kt","kotlin/text/numberCo
nversions.kt","kotlin/text/regex.kt","kotlin/text/stringsCode.kt","kotlin/text/utf8Encoding.kt","kotlin/time/DurationU

Open Source Used In DNAC Platform Arica 12992


Open Source Used In DNAC Platform Arica 12993
for (var i = 0; i < str.length; i++) {
    var code = str.charCodeAt(i);
    hash = (hash * 31 + code) | 0;  // Keep it 32-bit
}
return hash;

Kotlin.identityHashCode = getObjectHashCode;

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be
 * found in the LICENSE.txt file. 
 */

// Copyright 2009 The Closure Library Authors. All Rights Reserved.
//
// Licensed under the Apache License, Version 2.0 (the "License");
// you may not use this file except in compliance with the License. 
// You may obtain a copy of the License at
//
//    http://www.apache.org/licenses/LICENSE-2.0
//
// Unless required by applicable law or agreed to in writing, software
// distributed under the License is distributed on an "AS-IS" BASIS, 
// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

/**
 * Constructs a 64-bit two's-complement integer, given its low and high 32-bit
 * values as signed integers. See the from* functions below for more
 * convenient ways of constructing Longs.
 */
Kotlin.Long = function(low, high) {
  this.low_ = low | 0;  // force into 32 signed bits.

  this.high_ = high | 0;  // force into 32 signed bits.
};

Kotlin.Long.$metadata$ = {
    kind: "class",
    simpleName: "Long",
    interfaces:[]
};

NOTE: Common constant values

// NOTE: Common constant values

ZERO, ONE, NEG_ONE, etc. are defined below the
// from* methods on which they depend.

/**
 * A cache
 * of the Long representations of small integer values.
 * @type {!Object}
 * @private
 */
Kotlin.Long.IntCache_ = {};

/**
 * Returns a Long representing the given (32-bit) integer value.
 * The fractional part, if any, is rounded down towards zero.
 * Returns zero if this `Double` value is `NaN`, `Long.MIN_VALUE`
 * if it's less than `Long.MIN_VALUE`, `Long.MAX_VALUE`
 * if it's bigger than `Long.MAX_VALUE`. 
 * The number in question.
 */
Kotlin.Long.fromNumber = function(value) {
  if (isNaN(value)) {
    return Kotlin.Long.ZERO;
  } else if (value <= -Kotlin.Long.TWO_PWR_63_DBL_) {
    return Kotlin.Long.MIN_VALUE;
  } else if (value + 1 >= Kotlin.Long.TWO_PWR_63_DBL_) {
    return Kotlin.Long.MAX_VALUE;
  } else if (value < 0) {
    return Kotlin.Long.fromNumber(-value).negate();
  } else {
    return new Kotlin.Long((value % Kotlin.Long.TWO_PWR_32_DBL_) | 0);
  }
};

/**
 * Returns a Long representation of the given string, written using the given
 * radix. The textual representation of the Long.
 */
Kotlin.Long.fromString = function(str, opt_radix) {
  if (str.length == 0) {
    throw Error('number format error: empty string');
  }

  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  var matchedRadix = str.match(/^[0-9a-fA-F]+$/);
  if (!matchedRadix) {
    return null;
  }

  var radixValue = parseInt(str.replace(/([0-9a-fA-F]+)/, '16'), 16);
  return new Kotlin.Long((radixValue % Kotlin.Long.TWO_PWR_32_DBL_) | 0);
};
Kotlin.Long.fromString(str.substring(1), radix).negate();
  } else if (str.indexOf('-') >= 0) {
    throw Error('number format error: interior "-" character: ' + str);
  }

  // Do several (8) digits each time through the loop, so to minimize the calls to the very expensive emulated div.
  var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 8));
  if (size < 8) {
    var result = result.add(Kotlin.Long.fromNumber(value));
  } else {
    var result = result.multiply(radixToPower);
  }
  return result;

// NOTE: the compiler should inline these constant values below and then remove/un// these variables, so there should be no runtime penalty for these.

/**
 * Number used repeated below in calculations. This must appear before the first call to any from* function below.
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_16_DBL_ = 1 << 16;

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_24_DBL_ = 1 << 24;

/**
 * @type {number}
 * @private
 */

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_31_DBL_ = Kotlin.Long.TWO_PWR_32_DBL_ / 2;

/**
 * @type {number}
 * @private
 */

/**
 * @type {number}
 * @private
 */

/**
 * @type {number}
 * @private
 */
Kotlin.Long.TWO_PWR_63_DBL_ = Kotlin.Long.TWO_PWR_64_DBL_ / 2;

/** @type {!Kotlin.Long} */
Kotlin.Long.ZERO = Kotlin.Long.fromInt(0);

/** @type {!Kotlin.Long} */
Kotlin.Long.ONE = Kotlin.Long.fromInt(1);

/** @type {!Kotlin.Long} */
Kotlin.Long.MAX_VALUE = Kotlin.Long.fromBits(0xFFFFFFFF | 0, 0x7FFFFFFF | 0);

/** @type {!Kotlin.Long} */
Kotlin.Long.MIN_VALUE = Kotlin.Long.fromBits(0, 0x80000000 | 0);

/** @type {!Kotlin.Long} */
Kotlin.Long.TWO_PWR_24_ = Kotlin.Long.fromInt(1 << 24);

/** @return {number} The value, assuming it is a 32-bit integer. */
Kotlin.Long.prototype.toInt = function() {
  return this.low_;
};

/** @return {number} The closest floating-point representation to this value. */
Kotlin.Long.prototype.toNumber = function() {
  return this.high_ * Kotlin.Long.TWO_PWR_32_DBL_ +
  this.getLowBitsUnsigned();
};

/** @return {number} The 32-bit hashCode of this value. */
Kotlin.Long.prototype.hashCode = function() {
  return this.high_ ^ this.low_;
};

/**
 * @param {number=} opt_radix The radix in which the text should be written.
 * @return {string} The textual representation of this value.
 * @override
 */
Kotlin.Long.prototype.toString = function(opt_radix) {
  var radix = opt_radix || 10;
  if (radix < 2 || 36 < radix) {
    throw Error('radix out of range: ' + radix);
  }

  if (this.isZero()) {
    return '0';
  }

  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      // We need to change the Long value before it can be negated, so we remove the bottom-most digit in this base and then recurse to do the rest.
      var radixLong = Kotlin.Long.fromInt(0);
      var result = this.toString(radix);
      return '-' + result;
    }
  }

  // Do several (6) digits each time through the loop, so as to minimize the calls to the very expensive emulated div.
  var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 6));
  if (digits.length < 6) {
    var digits = 0 + digits;
    var result = 0 + digits + result;
  }

  var result = result.multiply(radixToPower);
  return result.add(Kotlin.Long.fromNumber(value));

  // Do several (8) digits each time through the loop, so as to minimize the calls to the very expensive emulated div.
  var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 8));
  if (size < 8) {
    var result = result.add(Kotlin.Long.fromNumber(value));
  } else {
    var result = result.multiply(radixToPower);
  }
  return result;
32-bits as an unsigned value. */

Kotlin.Long.prototype.getLowBitsUnsigned = function() {
    return (this.low_ >= 0) ?
      this.low_ : Kotlin.Long.TWO_PWR_32_DBL_ + this.low_;
};


/**
 * @return {number} Returns the number of bits needed to represent the absolute
 *     value of this Long.
 */
Kotlin.Long.prototype.getNumBitsAbs = function() {
  if (this.isNegative()) {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      return 64;
    } else {
      var val = this.high_ != 0 ? this.high_ : this.low_;
      for (var bit = 31; bit > 0; bit--) {
        if ((val & (1 << bit)) != 0) {
          break;
        }
      }
      return this.high_ != 0 ? bit + 33:
        bit + 1;
    }
  } else {
    return this.high_ == 0 ? 0:
      this.high_ != 0 ? bit + 33:
        bit + 1;
  }
};


/** @return {boolean} Whether this value is zero. */
Kotlin.Long.prototype.isZero = function() {
  return this.high_ == 0 && this.low_ == 0;
};


/** @return {boolean} Whether this value is negative. */
Kotlin.Long.prototype.isNegative = function() {
  return this.high_ < 0;
};


/** @return {boolean} Whether this value is odd. */
Kotlin.Long.prototype.isOdd = function() {
  return (this.low_ & 1) == 1;
};


/** @return {boolean} Whether this value is greater than the other.
 * @param {Kotlin.Long} other Long to compare against.
 * @return {boolean} Whether this Long does not equal the other.
 */
Kotlin.Long.prototype.notEqualsLong = function(other) {
    return (this.high_ != other.high_ || (this.low_ !=
      other.low_));
};


/** @return {!Kotlin.Long} The negation of this value. */
Kotlin.Long.prototype.negate = function() {
    if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      return Kotlin.Long.MIN_VALUE;
    } else {
      return this.not().add(Kotlin.Long.ONE);
    }
};


/**
 * Returns the sum of this and the given Long.
 * @param {Kotlin.Long} other Long to add to this one.
 * @return {!Kotlin.Long} The sum of this and the given
 *     Long.
 */
Kotlin.Long.prototype.add = function(other) {
  // Divide each number into 4 chunks of 16 bits, and
  // then sum the chunks.
  var a48 = this.high_ >>> 16;
  var a32 = this.high_ & 0xFFFF;
  var a16 = this.low_ >>> 16;
  var a00 = this.low_ & 0xFFFF;
  var b48 = other.high_ >>> 16;
  var b32 = other.high_ & 0xFFFF;
  var b16 = other.low_ >>> 16;
  var b00 = other.low_ & 0xFFFF;

  var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
  var d48 = 0, d32 = 0, d16 = 0, d00 = 0;

  c48 += a48 + b48;
  c32 += a32 + b32;
  c16 += a16 + b16;
  c00 += a00 + b00;

  d48 += a48 + b48;
  d32 += a32 + b32;
  d16 += a16 + b16;
  d00 += a00 + b00;

  return Kotlin.Long.fromBits(d16, d00, c16, c00, b16, b00, a16, a00);
};


/**
 * Returns the difference of this and the given Long.
 * @param {Kotlin.Long} other Long to subtract from this.
 * @return {!Kotlin.Long} The difference of this and the given
 *     Long.
 */
Kotlin.Long.prototype.subtract = function(other) {
  return this.add(other.negate());
};


/**
 * Returns the product of this and the given Long.
 * @param {Kotlin.Long} other Long to multiply with this.
 * @return {!Kotlin.Long} The product of this and the given
 *     Long.
 */
Kotlin.Long.prototype.multiply = function(other) {
  if (this.isZero()) {
    return Kotlin.Long.ZERO;
  } else if (other.isZero()) {
    return Kotlin.Long.ZERO;
  } else {
    return this.add(this.multiply(other));
  }
};


/**
 * Compares this Long with the given one.
 * @param {Kotlin.Long} other Long to compare against.
 * @return {number} 0 if they are the same, 1 if the this is greater, and -1
 *     if
 *     the given one is greater. */
Kotlin.Long.prototype.compare = function(other) {
  if (this.equalsLong(other)) {
    return 0;
  }

  var thisNeg = this.isNegative();
  var otherNeg = other.isNegative();
  if (thisNeg && !otherNeg) {
    return -1;
  }
  if (!thisNeg && otherNeg) {
    return 1;
  }

  // at this point, the signs are the
  // same, so subtraction will not overflow
  if (this.subtract(other).isNegative()) {
    return -1;
  } else {
    return 1;
  }
};
return other.isOdd() ? Kotlin.Long.MIN_VALUE : Kotlin.Long.ZERO;
  } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {
    return this.negate().multiply(other.negate());
  } else if (other.isNegative()) {
    return this.multiply(other.negate()).negate();
  } else // If both longs are small, use float multiplication
  if (this.lessThan(Kotlin.Long.TWO_PWR_24_)) {
    return Kotlin.Long.fromNumber(this.toNumber() * other.toNumber());
  } // Divide each long into 4 chunks of 16 bits, and then add up 4x4 products.
  // We can skip products that would overflow.
  var a48 = this.high_ >>> 16;
  var a32 = this.high_ & 0xFFFF;
  var a16 = this.low_ >>> 16;
  var a00 = this.low_ & 0xFFFF;
  var b48 = other.high_ >>> 16;
  var b32 = other.high_ & 0xFFFF;
  var b16 = other.low_ >>> 16;
  var b00 = other.low_ & 0xFFFF;
  var c48 = 0;
  var c32 = 0;
  var c16 = 0;
  var c00 = 0;
  var d48 = 0;
  var d32 = 0;
  var d16 = 0;
  var d00 = 0;
  c00 += a00 * b00;
  c16 += c00 >>> 16;
  c00 &= 0xFFFF;
  c16 += a16 * b00;
  c32 += c16 >>> 16;
  c16 &= 0xFFFF;
  c16 += a00 * b16;
  c32 += c16 >>> 16;
  c16 &= 0xFFFF;
  c16 += a32 * b00;
  c48 += c32 >>> 16;
  c32 &= 0xFFFF;
  c48 += a48 * b00 + a32 * b16 + a16 * b32 + a00 * b48;
  c48 &= 0xFFFF;
  return Kotlin.Long.fromBits((c16 << 16) | c00, (c48 << 16) | c32);
}

/**
 * Returns this Long divided by the given one.
 * @param {Kotlin.Long} other Long by which to divide.
 * @return {!Kotlin.Long} This Long divided by the given one.
 */
Kotlin.Long.prototype.div = function(other) {
  if (other.isZero()) {
    throw Error('division by zero');
  } else if (this.isZero()) {
    return Kotlin.Long.ZERO;
  } else if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
    if (other.equalsLong(Kotlin.Long.ONE) || other.equalsLong(Kotlin.Long.NEG_ONE)) {
      return Kotlin.Long.MIN_VALUE;  // recall that -MIN_VALUE == MIN_VALUE
    } else if (other.equalsLong(Kotlin.Long.MIN_VALUE)) {
      return Kotlin.Long.ZERO;
    } else // At this point, we have |other| >= 2, so |this/other| < |MIN_VALUE|.
      var halfThis = this.shiftRight(1);
      var approx = halfThis.div(other).shiftLeft(1);
      if (approx.equalsLong(Kotlin.Long.ZERO)) {
        return other.isNegative() ? Kotlin.Long.ONE : Kotlin.Long.NEG_ONE;
      } else {
        var rem = this.subtract(other.multiply(approx));
        var result = approx.add(rem.div(other));
        return result;
      }
  } else if (other.isNegative()) {
    return this.negate().div(other.negate());
  } else if (other.isNegative()) {
    return this.div(other.negate()).negate();
  }

  // Repeat the following until the remainder is less than other:  find a
  // floating-point that approximates remainder / other *from below*, add this
  // into the result, and subtract it from the remainder.  It is critical that
  // the approximate value is less than or equal to the real value so that the
  // remainder never becomes negative.
  var res = Kotlin.Long.ZERO;
  var rem = this;
  while (rem.greaterThanOrEqual(other)) {
    // Approximate the result of division. This may be a little greater or
    // smaller than the actual value.
    var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));
    // We will tweak the approximate result by changing it in the 48-th digit or
    // the smallest non-fractional digit, whichever is larger.
    var log2 = Math.ceil(Math.log(approx) / Math.LN2);
    var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);
    // Decrease the approximation until it is smaller than the remainder.
    Note
    // that if it is too large, the product overflows and is negative.
    var approxRem = rem.multiply(other);
    var approxRem = approxRem.multiply(other);
    var approxRem = approxRem.multiply(other);
    while (rem.isNegative()) { approxRem = approxRem.multiply(other);
      while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
      }
    }
    // Approximate the result of division. This may be a little greater or
    // smaller than the actual value.
    var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));
    // We will tweak the approximate result by changing it in the 48-th digit or
    // the smallest non-fractional digit, whichever is larger.
    var log2 = Math.ceil(Math.log(approx) / Math.LN2);
    var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);
    // Decrease the approximation until it is smaller than the remainder.
    Note
    // that if it is too large, the product overflows and is negative.
    var approxRem = rem.multiply(other);
    var approxRem = approxRem.multiply(other);
    var approxRem = approxRem.multiply(other);
    while (rem.isNegative()) { approxRem = approxRem.multiply(other);
      while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
      }
    }
    // Approximate the result of division. This may be a little greater or
    // smaller than the actual value.
    var approx = Math.max(1, Math.floor(rem.toNumber() / other.toNumber()));
    // We will tweak the approximate result by changing it in the 48-th digit or
    // the smallest non-fractional digit, whichever is larger.
    var log2 = Math.ceil(Math.log(approx) / Math.LN2);
    var delta = (log2 <= 48) ? 1 : Math.pow(2, log2 - 48);
    // Decrease the approximation until it is smaller than the remainder.
    Note
    // that if it is too large, the product overflows and is negative.
    var approxRem = rem.multiply(other);
    var approxRem = approxRem.multiply(other);
    var approxRem = approxRem.multiply(other);
    while (rem.isNegative()) { approxRem = approxRem.multiply(other);
      while (approxRem.isNegative() || approxRem.greaterThan(rem)) {
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
        approxRem = approxRem.multiply(other);
      }
    }
bitwise-NOT of this value. */

Kotlin.Long.prototype.not = function() {
  return Kotlin.Long.fromBits(~this.low_,
    ~this.high_);
};

Kotlin.Long.prototype.or = function(other) {
  return Kotlin.Long.fromBits(this.low_ +
    other.low_,
    this.high_ + other.high_);
};

Kotlin.Long.prototype.xor = function(other) {
  return Kotlin.Long.fromBits(this.low_ ^
    other.low_,
    this.high_ ^ other.high_);
};

Kotlin.Long.prototype.shl = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var low = this.low_;
    if (numBits < 32) {
      var high = this.high_;
      return Kotlin.Long.fromBits(
        low << numBits,
        (high << numBits) | (low >>> (32 - numBits)));
    } else {
      return Kotlin.Long.fromBits(0,
        low << (numBits - 32));
    }
  }
};

Kotlin.Long.prototype.shr = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(
        (low >>> numBits) | (high << (32 - numBits)),
        high >> numBits);
    } else if (numBits == 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >>
        (numBits - 32), 0);
    }
  }
};

Kotlin.Long.prototype.shru = function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(
        (low >>> numBits) | (high << (32 - numBits)),
        high >>> numBits);
    } else if (numBits == 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >>
        (numBits - 32), 0);
    }
  }
};

Kotlin.defineModule = function(id, declaration) {
};

Kotlin.defineInlineFunction = function(tag, fun) {
  return fun;
};

Kotlin.wrapFunction = function(fun) {
    var f = function() {
        f = fun();
        return f.apply(this, arguments);
    };
    return function() {
        return f.apply(this, arguments);
    };
};

Kotlin.orNull = function(fn) {
    return function(object) {
        return object || fn(object);
    };
};

Kotlin.isTypeOf = function(type) {
    return function(object) {
        return typeof object === type;
    };
};
value) { 
    var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);
    if (propertyDescriptor !== null && propertyDescriptor.set !== null) { 
        propertyDescriptor.set.call(thisObject, value);
        return;
    }

    propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject, propertyName);
    if (propertyDescriptor !== null && "value" in propertyDescriptor) { 
        thisObject[propertyName] = value;
        return;
    }

    Kotlin.callSetter(thisObject, Object.getPrototypeOf(klass), propertyName, value);
}

function isInheritanceFromInterface(ctor, iface) {
    if (ctor === iface) return true;

    var metadata = ctor.$metadata$;
    if (metadata != null) {
        var interfaces = metadata.interfaces;
        for (var i = 0; i < interfaces.length; i++) { 
            if (isInheritanceFromInterface(interfaces[i], iface)) { 
                return true;
            }
        }
    }

    var superPrototype = ctor.prototype != null ? Object.getPrototypeOf(ctor.prototype) : null;
    var superConstructor = superPrototype != null ? superPrototype.constructor : null;
    return superConstructor != null && isInheritanceFromInterface(superConstructor, iface);
}

/**
 * @param {*} object
 * @param {Function|Object} klass
 * @returns {Boolean}
 */
Kotlin.isType = function (object, klass) {
    if (klass === Object) {
        switch (typeof object) {
            case "string":
            case "number":
            case "boolean":
            case "function":
                return true;
            default:
                return object instanceof Object;
        }
    }

    if (object == null || klass == null || (typeof object !== 'object' && typeof object !== 'function')) {
        return false;
    }

    if (typeof klass === "function" && object instanceof klass) {
        return true;
    }

    var proto = Object.getPrototypeOf(klass);
    var constructor = proto != null ? proto.constructor : null;
    if (constructor != null && "$metadata$" in constructor) {
        var klassMetadata = constructor.$metadata$;
        if (klassMetadata.kind === Kotlin.Kind.OBJECT) {
            return object === klass;
        }
    }

    var klassMetadata = klass.$metadata$;

    // In WebKit (JavaScriptCore) for some interfaces from DOM typeof returns "object", nevertheless they can be used in RHS of instanceof
    if (klassMetadata == null) {
        return object instanceof klass;
    }

    if (klassMetadata.kind === Kotlin.Kind.INTERFACE && object.constructor != null) {
        return isInheritanceFromInterface(object.constructor, klass);
    }

    return false;
}

Kotlin.isNumber = function (a) {
    return typeof a == "number" || a instanceof Kotlin.Long;
};

Kotlin.isChar = function (value) {
    return value instanceof Kotlin.BoxedChar;
};

Kotlin.isComparable = function (value) {
    var type = typeof value;
    return type === "string" || type === "boolean" || Kotlin.isNumber(value) || Kotlin.isType(value, Kotlin.kotlin.Comparable);
};

Kotlin.isCharSequence = function (value) {
    return typeof value === "string" || Kotlin.isType(value, Kotlin.kotlin.CharSequence);
};

private var _ordinal: Int = 0

override fun compareTo(other: Enum<T>) = ordinal.compareTo(other.ordinal)

override fun equals(other: Any?) = this === other

override fun toString() = String("\n    \n  @JsName("\nprivate var _name: String = "\n
var private _name: String = "\n
override fun get() = _name

override fun compare(other: Enum<T>) = ordinal.compareTo(other.ordinal)

override fun equals(other: Any?) = this === other

override fun hashCode() = js("Kotlin.identityHashCode")

override fun companionObject() = "\n
override fun companionObject() = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = "\n
private var _name: String = ""
get(0)

* Returns 1st *element* from the array.

* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component1(): Short {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component1(): Int {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component1(): Long {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component1(): Float {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component1(): Double {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component1(): Boolean {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component1(): Char {
  return
  get(0)
}

* Returns 1st *element* from the array.
* If the size of this array is less than 1, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component2(): Byte {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component2(): Short {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component2(): Int {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component2(): Long {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component2(): Float {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component2(): Double {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component2(): Boolean {
  return
  get(1)
}

* Returns 2nd *element* from the array.
* If the size of this array is less than 2, throws an
  [IndexOutOfBoundsException] except in Kotlin/JS
* where the behavior is unspecified.

*\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component2(): Char {
  return
  get(1)
}
get(1) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun <T> Array<T>.component3(): T { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun ByteArray.component3(): Byte { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun ShortArray.component3(): Short { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun IntArray.component3(): Int { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun LongArray.component3(): Long { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun FloatArray.component3(): Float { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun DoubleArray.component3(): Double { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun BooleanArray.component3(): Boolean { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun CharArray.component3(): Char { return
get(2) // Returns 3rd *element* from the array. *If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun ByteArray.component4(): Byte { return
get(3) // Returns 4th *element* from the array. *If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun ShortArray.component4(): Short { return
get(3) // Returns 4th *element* from the array. *If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun IntArray.component4(): Int { return
get(3) // Returns 4th *element* from the array. *If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun LongArray.component4(): Long { return
get(3) // Returns 4th *element* from the array. *If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun FloatArray.component4(): Float { return
get(3) // Returns 4th *element* from the array. *If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS* *where the behavior is unspecified.*

//@kotlin.internal.InlineOnly//public inline operator fun DoubleArray.component4(): Double { return
get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.

get(3) * Returns 4th *element* from the array.
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun <T> Array-out T>.elementAt(index: Int): T

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun ByteArray.elementAt(index: Int): Byte

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun ShortArray.elementAt(index: Int): Short

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun IntArray.elementAt(index: Int): Int

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun LongArray.elementAt(index: Int): Long

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun FloatArray.elementAt(index: Int): Float

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun DoubleArray.elementAt(index: Int): Double

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun BooleanArray.elementAt(index: Int): Boolean

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public fun CharArray.elementAt(index: Int): Char

- Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.

*Public inline fun <T> Array-out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun ByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Byte): Byte

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun ShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Short): Short

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*Public inline fun CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char

- Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the
 * result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun
BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
BooleanArray.find(predicate: (Boolean) -> Boolean): Boolean? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun
CharArray.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
}
element was found.
*/
@kotlin.internal.InlineOnly
public inline fun LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun StringArray.find(predicate: (String) -> Boolean): String? {
    return firstOrNull(predicate)
}

/**
 * Returns the first element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.find(predicate: (Char) -> Boolean): Char? {
    return firstOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.findLast(predicate: (Double) -> Boolean): Double? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun StringArray.findLast(predicate: (String) -> Boolean): String? {
    return lastOrNull(predicate)
}

/**
 * Returns the last element matching the given [predicate], or ‘null’ if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun <T> Array<out T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ByteArray.first(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ShortArray.first(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun IntArray.first(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun LongArray.first(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun FloatArray.first(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun DoubleArray.first(): Double {
    if (isEmpty())
        throwNoSuchElementException("Array is empty.")
    return this[0]
}
public fun BooleanArray.first(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun charArray.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun <T> Array<out T>.first(predicate: (T) -> Boolean): T {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ByteArray.first(predicate: (Byte) -> Boolean): Byte {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ShortArray.first(predicate: (Short) -> Boolean): Short {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun IntArray.first(predicate: (Int) -> Boolean): Int {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun LongArray.first(predicate: (Long) -> Boolean): Long {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun CharArray.first(predicate: (Char) -> Boolean): Char {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

// Additional code for Array.firstOrNull() and other similar methods
matching the given [predicate], or 'null' if element was not found.

 */
public inline fun <T> Array<out T>.firstOrNull(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun ByteArray.firstOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun ShortArray.firstOrNull(predicate: (Short) -> Boolean): Short? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun IntArray.firstOrNull(predicate: (Int) -> Boolean): Int? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun LongArray.firstOrNull(predicate: (Long) -> Boolean): Long? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun FloatArray.firstOrNull(predicate: (Float) -> Boolean): Float? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun DoubleArray.firstOrNull(predicate: (Double) -> Boolean): Double? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun BooleanArray.firstOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns the first element matching the given [predicate], or 'null' if element was not found.
 */
public inline fun CharArray.firstOrNull(predicate: (Char) -> Boolean): Char? {
    for (element in this) if (predicate(element)) return element
    return null
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.getOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.getOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
element matching the given [predicate], or -1 if the array does not contain such element.

*`public inline fun`* 

**ByteArray.indexOfFirst(predicate: (Byte) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**ShortArray.indexOfFirst(predicate: (Short) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int**

```java
for (index in indices.reversed()) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int**

```java
for (index in indices.reversed()) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**LongArray.indexOfLast(predicate: (Long) -> Boolean): Int**

```java
for (index in indices.reversed()) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int**

```java
for (index in indices.reversed()) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**IntArray.indexOfFirst(predicate: (Int) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```

*`public inline fun`* 

**CharArray.indexOfLast(predicate: (Char) -> Boolean): Int**

```java
for (index in indices) {
    if (predicate(this[index])) {
        return index
    } 
} 
return -1
```
element matching the given [predicate], or -1 if the array does not contain such element.\n*/
public inline fun CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index]))
            return index
    }
    return -1
}

//
public inline fun CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index]))
            return index
    }
    return -1
}

/**
 * Returns the last element.
 */
public inline fun CharArray.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun ByteArray.last(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun ShortArray.last(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun IntArray.last(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun LongArray.last(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun FloatArray.last(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun DoubleArray.last(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element.
 */
public inline fun BooleanArray.last(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]!!
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun CharArray.last(predicate: (Char) -> Boolean): Char {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun ByteArray.last(predicate: (Byte) -> Boolean): Byte {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun ShortArray.last(predicate: (Short) -> Boolean): Short {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun IntArray.last(predicate: (Int) -> Boolean): Int {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun LongArray.last(predicate: (Long) -> Boolean): Long {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun FloatArray.last(predicate: (Float) -> Boolean): Float {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun DoubleArray.last(predicate: (Double) -> Boolean): Double {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 */
public inline fun BooleanArray.last(predicate: (Boolean) -> Boolean): Boolean {
    for (index in indices.reversed()) {
        val element = this[index]
        if (predicate(element))
            return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
[NoSuchElementException] if no such element is found.

```kotlin
/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun Array<out T>.last(predicate: (T) -> Boolean): T? {
    if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.
"
)
}

/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun <U> Array<out U>.lastOf(element: U): U? {
    if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.
"
)
}
```

Open Source Used In DNAC Platform Arica 13015
```kotlin
/*
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public fun CharArray.lastOrNull(): Char? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.lastOrNull(predicate: (T) -> Boolean): T? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or 'null' if no such
 * element was found.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    var lastElement = this[0]
    for (l in this.indices) {
        if (predicate(lastElement)) return lastElement
        lastElement = this[l]
    }
    return null
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.random(): T? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ByteArray.random(): Byte? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ShortArray.random(): Short? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntArray.random(): Int? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongArray.random(): Long? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun FloatArray.random(): Float? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun DoubleArray.random(): Double? {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun BooleanArray.random(): Boolean? {
    return random(Random)
}
```
from this array.

```kotlin
@throws NoSuchElementException if this array is empty.

public inline fun CharArray.random(): Char {
    return random(Random)\n    @throws NoSuchElementException if this array is empty.
    return get(random.nextInt(size))\n}

public fun <T> Array<out T>.random(random: Random): T {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun ShortArray.random(random: Random): Short {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun IntArray.random(random: Random): Int {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun LongArray.random(random: Random): Long {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun FloatArray.random(random: Random): Float {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun DoubleArray.random(random: Random): Double {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun BooleanArray.random(random: Random): Boolean {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun ByteArray.random(random: Random): Byte {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}

public fun ByteArray.randomOrNull(): Byte? {
    \n    if (isEmpty()) \n    throw NoSuchElementException("Array is empty.")\n    return get(random.nextInt(size))\n}
```

---

Open Source Used In DNAC Platform Arica 13017
from this array, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public inline fun FloatArray.randomOrNull(): Float? {\n    return randomOrNull(Random)\n}\n/**\n * Returns a random element from this array, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public inline fun DoubleArray.randomOrNull(): Double? {\n    return randomOrNull(Random)\n}\n/**\n * Returns a random element from this array, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public inline fun BooleanArray.randomOrNull(): Boolean? {\n    return randomOrNull(Random)\n}\n/**\n * Returns a random element from this array, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public inline fun CharArray.randomOrNull(): Char? {\n    return randomOrNull(Random)\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun \n<T> Array\out T>.randomOrNull(random: Random): T? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun ByteArray.randomOrNull(random: Random): Byte? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun ShortArray.randomOrNull(random: Random): Short? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun IntArray.randomOrNull(random: Random): Int? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun LongArray.randomOrNull(random: Random): Long? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun FloatArray.randomOrNull(random: Random): Float? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun DoubleArray.randomOrNull(random: Random): Double? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun BooleanArray.randomOrNull(random: Random): Boolean? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns a random element from this array using the specified source of randomness, or ‘null’ if this array is empty.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun CharArray.randomOrNull(random: Random): Char? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\public fun ShortArray.randomOrNull(): Short? {\n    if (isEmpty()) return null\n    return get(random.nextInt(size))\n}\n/**\n * Returns the single element, or throws an exception if the array is empty or has more than one element.

Open Source Used In DNAC Platform Arica  13018
IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun IntArray.single(): Int {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single
 * element, or throws an exception if the array is empty or has more than one element.
 */
public fun LongArray.single(): Long {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun FloatArray.single(): Float {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element, or throws an exception if the array is empty or has more than one element.
 */
public fun DoubleArray.single(): Double {
    return when (size) {
        0 -> throw NoSuchElementException("Array is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Array has more than one element.")
    }
}

/**
 * Returns the single element
 * matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Array<out T>.single(predicate: (T) -> Boolean): T {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns the single element matching
 * the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ByteArray.single(predicate: (Byte) -> Boolean): Byte {
    var single: Byte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Byte
}

/**
 * Returns the single element matching
 * the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ShortArray.single(predicate: (Short) -> Boolean): Short {
    var single: Short? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Short
}

/**
 * Returns the single element matching
 * the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Int
}

/**
 * Returns the single element matching
 * the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST")
    return single as Long
}
contains more than one matching element.

```kotlin
open fun <T> Array<out T>.singleOrNull(): T? {
    return if (size == 1) this[0] else null
}

public inline fun ByteArray.singleOrNull(): Byte? {
    return if (size == 1) this[0] else null
}

public inline fun ShortArray.singleOrNull(): Short? {
    return if (size == 1) this[0] else null
}

public inline fun IntArray.singleOrNull(): Int? {
    return if (size == 1) this[0] else null
}

public inline fun LongArray.singleOrNull(): Long? {
    return if (size == 1) this[0] else null
}

public inline fun FloatArray.singleOrNull(): Float? {
    return if (size == 1) this[0] else null
}

public inline fun DoubleArray.singleOrNull(): Double? {
    return if (size == 1) this[0] else null
}

public inline fun BooleanArray.singleOrNull(): Boolean? {
    return if (size == 1) this[0] else null
}

public inline fun CharArray.singleOrNull(): Char? {
    return if (size == 1) this[0] else null
}
```

}
ByteArray.singleOrNull(predicate: (Byte) -> Boolean): Byte? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun ShortArray.singleOrNull(predicate: (Short) -> Boolean): Short? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun IntArray.singleOrNull(predicate: (Int) -> Boolean): Int? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun LongArray.singleOrNull(predicate: (Long) -> Boolean): Long? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun FloatArray.singleOrNull(predicate: (Float) -> Boolean): Float? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun DoubleArray.singleOrNull(predicate: (Double) -> Boolean): Double? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
public inline fun BooleanArray.singleOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    var single = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun <T> Array<out T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ByteArray.drop(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun ShortArray.drop(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun IntArray.drop(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun LongArray.drop(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun FloatArray.drop(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun DoubleArray.drop(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun BooleanArray.drop(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.drop
 */
public fun CharArray.drop(n: Int): List<Character> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}
n).coerceAtLeast(0))

// Returns a list containing all elements except last [n] elements.

public fun BooleanArray.dropLast(n: Int):
List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws
 * DiameterException if [n] is negative.
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public fun CharArray.dropLast(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except last elements that satisfy the given 
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun <T> Array<out T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {
    var yielding = false
    val list = ArrayList<Byte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {
    var yielding = false
    val list = ArrayList<Short>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {
    var yielding = false
    val list = ArrayList<Int>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {
    var yielding = false
    val list = ArrayList<Long>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun

// Returns a list containing all elements except last elements that satisfy the given
// [predicate].

public fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun

// Returns a list containing all elements except last elements that satisfy the given
// [predicate].

public fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun

// Returns a list containing all elements except last elements that satisfy the given
// [predicate].

public fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except first elements that satisfy the given
 * [predicate].
 * 
 * @sample
 * samples.collections.Collections.Transformations.drop
 */
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
yielding = true
    return list
}

* Returns a list containing all elements except first elements that satisfy the given [predicate].

```kotlin
public inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {
    var yielding = false
    val list = ArrayList<Byte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
public inline fun CharArray.dropWhile(predicate: (Char) -> Boolean): List<Char> {
    var yielding = false
    val list = ArrayList<Char>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
public inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {
    var yielding = false
    val list = ArrayList<Int>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
public inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {
    var yielding = false
    val list = ArrayList<Short>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```
fun LongArray.filter(predicate: (Long) -> Boolean): List<Long> { return filterTo(ArrayList<Long>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Filtering.filter
 */
public inline fun LongArray.filter(predicate: (Long) -> Boolean): List<Long> { return filterTo(ArrayList<Long>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Filtering.filter
 */
public inline fun FloatArray.filter(predicate: (Float) -> Boolean): List<Float> { return filterTo(ArrayList<Float>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Filtering.filter
 */
public inline fun DoubleArray.filter(predicate: (Double) -> Boolean): List<Double> { return filterTo(ArrayList<Double>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Filtering.filter
 */
public inline fun BooleanArray.filter(predicate: (Boolean) -> Boolean): List<Boolean> { return filterTo(ArrayList<Boolean>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Filtering.filter
 */
public inline fun CharArray.filter(predicate: (Char) -> Boolean): List<Char> { return filterTo(ArrayList<Char>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Array<out T>.filterIndexed(predicate: (index: Int, T) -> Boolean): List<T> { return filterIndexedTo(ArrayList<T>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun ByteArray.filterIndexed(predicate: (index: Int, Byte) -> Boolean): List<Byte> { return filterIndexedTo(ArrayList<Byte>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun ShortArray.filterIndexed(predicate: (index: Int, Short) -> Boolean): List<Short> { return filterIndexedTo(ArrayList<Short>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun IntArray.filterIndexed(predicate: (index: Int, Int) -> Boolean): List<Int> { return filterIndexedTo(ArrayList<Int>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun LongArray.filterIndexed(predicate: (index: Int, Long) -> Boolean): List<Long> { return filterIndexedTo(ArrayList<Long>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun FloatArray.filterIndexed(predicate: (index: Int, Float) -> Boolean): List<Float> { return filterIndexedTo(ArrayList<Float>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun DoubleArray.filterIndexed(predicate: (index: Int, Double) -> Boolean): List<Double> { return filterIndexedTo(ArrayList<Double>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun BooleanArray.filterIndexed(predicate: (index: Int, Boolean) -> Boolean): List<Boolean> { return filterIndexedTo(ArrayList<Boolean>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun CharArray.filterIndexed(predicate: (index: Int, Char) -> Boolean): List<Char> { return filterIndexedTo(ArrayList<Char>(), predicate) * Returns a list containing only elements matching the given [predicate].
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * 
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
public inline fun <T, C : MutableCollection<in T>> Array<out T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C { return filterIndexedTo(destination, predicate) * Appends all elements matching the given [predicate] to the given [destination].
}
index, element ->
if (predicate(index, element)) destination.add(element)
return destination

@public
@inline fun <C : Collection<in T>> T.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed {
        index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public fun <C : Collection<in T>> T.filterNotTo(destination: C): C {
    for (element in this)
        if (!element is R)
            destination.add(element)
    return destination
}

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 */
public fun <R> Array<out T>.filterIsInstanceTo(destination: List<R>): List<R> {
    return filterIsInstanceTo(ArrayList<R>())
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public fun <R, C : Collection<in R>> Array<out T>.filterIsInstanceTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    for (element in this)
        if (element is R)
            destination.add(element)
    return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 */
public fun <T> Array<out T>.filterNot(predicate: (T) -> Boolean): List<T> {
    return filterNotTo(ArrayList<T>())
}
predicate)

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun ByteArray.filterNot(predicate: (Byte) -> Boolean): List<Byte> {
    return filterNotTo(ArrayList<Byte>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun ShortArray.filterNot(predicate: (Short) -> Boolean): List<Short> {
    return filterNotTo(ArrayList<Short>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun IntArray.filterNot(predicate: (Int) -> Boolean): List<Int> {
    return filterNotTo(ArrayList<Int>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun LongArray.filterNot(predicate: (Long) -> Boolean): List<Long> {
    return filterNotTo(ArrayList<Long>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun FloatArray.filterNot(predicate: (Float) -> Boolean): List<Float> {
    return filterNotTo(ArrayList<Float>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun DoubleArray.filterNot(predicate: (Double) -> Boolean): List<Double> {
    return filterNotTo(ArrayList<Double>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun BooleanArray.filterNot(predicate: (Boolean) -> Boolean): List<Boolean> {
    return filterNotTo(ArrayList<Boolean>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filter
 */
public inline fun CharArray.filterNot(predicate: (Char) -> Boolean): List<Char> {
    return filterNotTo(ArrayList<Char>(), predicate)
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 * 
 * @sample collections.Filters.Filtering.filterNotNull
 */
public fun <T : Any> Array<out T?>.filterNotNull(): List<T> {
    return filterNotNullTo(ArrayList<T>())
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Array<out T?>.filterNotTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Short>> ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Int>> IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Long>> LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Float>> FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Double>> DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}
/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Char>> CharArray.filterNotTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Short>> ShortArray.filterNotTo(destination: C, predicate: (Short) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Int>> IntArray.filterNotTo(destination: C, predicate: (Int) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Long>> LongArray.filterNotTo(destination: C, predicate: (Long) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Float>> FloatArray.filterNotTo(destination: C, predicate: (Float) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : MutableCollection<in Double>> DoubleArray.filterNotTo(destination: C, predicate: (Double) -> Boolean): C {
    for (element in this)
    if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun <T> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun ShortArray.slice(indices: IntRange): List<Short> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in thespecified [indices] range.
 */
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
indices in the specified [indices] range.

\*\n
//} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun BooleanArray.slice(indices: IntRange): List<Boolean> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun DoubleArray.slice(indices: IntRange): List<Double> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun ByteArray.slice(indices: IntRange): List<Byte> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun CharArray.slice(indices: IntRange): List<Char> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun ShortArray.slice(indices: IntRange): List<Short> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun IntArray.slice(indices: IntRange): List<Int> { //} if (indices.isEmpty()) return listOf(0)
  return copyOfRange(indices.start, indices.endInclusive + 1).asList().get(0)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Char>(size)
  return ArrayList<Char>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Boolean>(size)
  return ArrayList<Boolean>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Double>(size)
  return ArrayList<Double>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Long>(size)
  return ArrayList<Long>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Int>(size)
  return ArrayList<Int>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Short>(size)
  return ArrayList<Short>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Byte>(size)
  return ArrayList<Byte>(size)
//} * Returns a list containing elements at specified indices.
\*\n
fun ArrayList<Byte>(size)
LongArray.sliceArray(indices: Collection<Int>): LongArray { val result = LongArray(indices.size) var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

Array<T>.sliceArray(indices: IntRange): Array<T> {
    if (indices.isEmpty()) return copyOfRange(0, 0)
    return copyOfRange(indices.start, indices.endInclusive + 1)
}

* @returns an array containing elements at indices in the specified [indices] range.

BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {
    val result = BooleanArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

CharArray.sliceArray(indices: Collection<Int>): CharArray {
    val result = CharArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

DoubleArray.sliceArray(indices: Collection<Int>): DoubleArray {
    val result = DoubleArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

FloatArray.sliceArray(indices: Collection<Int>): FloatArray {
    val result = FloatArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

IntArray.sliceArray(indices: Collection<Int>): IntArray {
    val result = IntArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns an array containing elements at indices in the specified [indices] range.

LongArray.sliceArray(indices: Collection<Int>): LongArray {
    val result = LongArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

* @returns a list containing first [n] elements.

* @throws IllegalArgumentException if [n] is negative.

Array<T> take(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

* @returns a list containing first [n] elements.

* @throws IllegalArgumentException if [n] is negative.

Open Source Used In DNAC Platform Arica  13030
samples.collections.Collections.Transformations.take\n *\n *\n
fun ShortArray.take(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Short>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun IntArray.take(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Int>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun ByteArray.take(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[size - 1])\n    var count = 0\n    val list = ArrayList<Byte>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun CharArray.take(n: Int): List<Char> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Char>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun BooleanArray.take(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Boolean>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun DoubleArray.take(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Double>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun FloatArray.take(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Float>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}

fun LongArray.take(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[0])\n    var count = 0\n    val list = ArrayList<Long>()\n    for (item in this) {
        list.add(item)\n        if (++count == n) break\n    }\n    return list
}
size\n if (n >= size) return toList()\n if (n == 1) return listOf(this[size - 1])\n val list = ArrayList<Char>(n)\n for (index in size - n until size)\n    list.add(this[index])\n return list\n\n@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public inline fun <T> takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun <T> Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.collections.Collections.Transformations.take

/*
 * @sample samples.collections.Collections.Transformations.take
 */
public fun Array<out T>.takeLast(n: Int): List<T> {
    for (index in size - n until size)\n        list.add(this[index])\n    return list\n}\n
@sample samples.c
LongArray.takeLastWhile(predicate: (Long) -> Boolean): List<Long> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

ByteArray.takeLastWhile(predicate: (Byte) -> Boolean): List<Byte> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

CharArray.takeLastWhile(predicate: (Char) -> Boolean): List<Char> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

DoubleArray.takeLastWhile(predicate: (Double) -> Boolean): List<Double> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

FloatArray.takeLastWhile(predicate: (Float) -> Boolean): List<Float> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

IntArray.takeLastWhile(predicate: (Int) -> Boolean): List<Int> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

LongArray.takeLastWhile(predicate: (Long) -> Boolean): List<Long> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

ShortArray.takeLastWhile(predicate: (Short) -> Boolean): List<Short> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.

StackTraceElementArray.takeLastWhile(predicate: (StackTraceElement) -> Boolean): List<StackTraceElement> {
for (index in lastIndex downTo 0) {
if (!predicate(this[index])) {
return drop(index + 1)
}
return toList().unmodifiableList() * Returns a list containing last elements satisfying the given [predicate].
}
* @sample samples.collections.Collections.Transformations.takeLastWhile function.
return list

*/

Reverse elements in the array in-place.

 */

public fun <T> Array<T>.reverse(): Unit {

val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint)

val tmp = this[index]

this[index] = this[reverseIndex]

this[reverseIndex] = tmp

reverseIndex--

*/

Reverse elements in the array in-place.

 */

public fun intArray.<T>Array.<T>.reverse(): Unit {

val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint)

val tmp = this[index]

this[index] = this[reverseIndex]

this[reverseIndex] = tmp

reverseIndex--

*/

Reverse elements in the array in-place.

 */

public fun longArray.<T>Array.<T>.reverse(): Unit {

val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint)

val tmp = this[index]

this[index] = this[reverseIndex]

this[reverseIndex] = tmp

reverseIndex--

*/

Reverse elements in the array in-place.

 */

public fun shortArray.<T>Array.<T>.reverse(): Unit {

val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint)

val tmp = this[index]

this[index] = this[reverseIndex]

this[reverseIndex] = tmp

reverseIndex--

*/

Reverse elements in the array in-place.

 */

public fun byteArray.<T>Array.<T>.reverse(): Unit {

val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint)

val tmp = this[index]

this[index] = this[reverseIndex]

this[reverseIndex] = tmp

reverseIndex--

*/

Reverse elements in the array in-place.

 */

public fun <T> Array.<T>.reversedArray(): Array<T> {

if (isEmpty()) return this

val lastIndex = lastIndex

for (i in 0..lastIndex)

result[lastIndex - i] = this[i]

return
according to their natural sort order.

```kotlin
public fun ByteArray.sorted(): List<Byte> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun ShortArray.sorted(): List<Short> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun IntArray.sorted(): List<Int> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun LongArray.sorted(): List<Long> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun FloatArray.sorted(): List<Float> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun DoubleArray.sorted(): List<Double> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun CharArray.sorted(): List<Char> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 */
public fun ByteArray.sortedArray(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun ShortArray.sortedArray(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun IntArray.sortedArray(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun LongArray.sortedArray(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun FloatArray.sortedArray(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted according to their natural sort order.
 */
public fun DoubleArray.sortedArray(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun ByteArray.sortedArrayDescending(): ByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun ShortArray.sortedArrayDescending(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun IntArray.sortedArrayDescending(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun LongArray.sortedArrayDescending(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}

/**
 * Returns an array with all elements of this array sorted descending according to their natural sort order.
 */
public fun DoubleArray.sortedArrayDescending(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortWith(reverseOrder()) }
}
```

Open Source Used In DNAC Platform Arica 13036
CharArray.sortedArrayDescending(): CharArray

if (isEmpty()) return this
return this.copyOf().apply {
 sortDescending() }

* Returns an array with all elements of this array sorted according to the specified comparator. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

Array.sortWith(compareWith(comparator: Comparator<T>)): Array<T>

if (isEmpty()) return this
return this.copyOf().apply {
 sortWith(compareWith(comparator: Comparator<T>)) }

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte>

return sortedWith(compareWith(selector: Comparator<Byte>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short>

return sortedWith(compareWith(selector: Comparator<Short>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int>

return sortedWith(compareWith(selector: Comparator<Int>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long>

return sortedWith(compareWith(selector: Comparator<Long>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float>

return sortedWith(compareWith(selector: Comparator<Float>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double>

return sortedWith(compareWith(selector: Comparator<Double>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.

CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char>

return sortedWith(compareWith(selector: Comparator<Char>()))

* Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.
Open Source Used In DNAC Platform Arica 13039
Boolean\n
* Returns ‘true’ if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n */\n@SinceKotlin("1.1")\npublic expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean\n
/**
 * Returns ‘true’ if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n */\n@SinceKotlin("1.1")\npublic expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the same elements in the same order.\n * The elements are compared for equality with the [equals][Any.equals] function.\n * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.\n */\n@SinceKotlin("1.1")\npublic expect infix fun CharArray.contentEquals(other: CharArray): Boolean

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun <T> Array<out T>.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun ByteArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun ShortArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun IntArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun LongArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun FloatArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun DoubleArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun BooleanArray.contentHashCode(): Int

/**
 * Returns a hash code based on the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun CharArray.contentHashCode(): Int

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun <T> Array<out T>.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun ByteArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun ShortArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun IntArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun LongArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun FloatArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun DoubleArray.contentToString(): String

/**
 * Returns a string representation of the contents of the specified array as if it is \[List\].\n */\n@SinceKotlin("1.1")\npublic expect fun BooleanArray.contentToString(): String

**Open Source Used In DNAC Platform Arica 13040**
IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun LongArray.copyInto(destination: LongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): LongArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @return the destination array.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun StringArray.copyInto(destination: StringArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): StringArray

* @param destination the array to copy to.
* @param destinationOffset the position in the destination array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination array starting at the specified destinationOffset, or when that index is out of the destination array indices range.

* @return the destination array.

@SinceKotlin("1.3")
public expect fun CharSequence.copyInto(destination: CharSequence, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharSequence

* @param destination the sequence to copy to.
* @param destinationOffset the position in the destination sequence to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this sequence by default.

* @throws IndexOutOfBoundsException or IllegalArgumentException when startIndex or endIndex is out of range of this array indices or when startIndex > endIndex.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the destination sequence starting at the specified destinationOffset, or when that index is out of the destination sequence indices range.

* @return the destination sequence.
samples.collections.Arrays.CopyOfOperations.copyOf

*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\n\npublic expect fun <T> Array<T>.copyOf():
\nArray<T>\n\n**\n\n* Returns new array which is a copy of the original array.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n
public expect fun ByteArray.copyOf():
\nByteArray\n\n**\n\n* Returns new array which is a copy of the original array.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n
public expect fun IntArray.copyOf():
\nIntArray\n\n**\n\n* Returns new array which is a copy of the original array.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n
public expect fun LongArray.copyOf():
\nLongArray\n\n**\n\n* Returns new array which is a copy of the original array.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n
public expect fun FloatArray.copyOf():
\nFloatArray\n\n**\n\n* Returns new array which is a copy of the original array.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.copyOf\n
public expect fun CharArray.copyOf():
\nCharArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
public expect fun ByteArray.copyOf(newSize: Int):
\nByteArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
public expect fun ShortArray.copyOf(newSize: Int):
\nShortArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
public expect fun IntArray.copyOf(newSize: Int):
\nIntArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
public expect fun LongArray.copyOf(newSize: Int):
\nLongArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
public expect fun FloatArray.copyOf(newSize: Int):
\nFloatArray\n\n**\n\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with zero values if necessary.\n* \n* @sample
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n
Open Source Used In DNAC Platform Arica  13043
*\n**public expect fun DoubleArray.copyOf(newSize: Int): DoubleArray\n**\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with `false` values if necessary.\n* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `false` values.\n* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\n\npublic expect fun BooleanArray.copyOf(newSize: Int): BooleanArray\n**\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with null char (`\u0000`) values if necessary.\n* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with null char (`\u0000`) values.\n* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\n\npublic expect fun CharArray.copyOf(newSize: Int): CharArray\n**\n* Returns new array which is a copy of the original array, resized to the given [newSize].\n* The copy is either truncated or padded at the end with `null` values if necessary.\n* - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].\n* - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with `null` values.\n* @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\n\npublic expect fun <T> Array<T>.copyOf(newSize: Int): Array<T?>\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T>\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray\n**\n*\n@Suppress("NO_ACTUAL_FOR_EXPECT")\npublic expect fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray\n**\n* Fills this array

Open Source Used In DNAC Platform Arica  13044
or its subrange with the specified \[element\] value.\n * **@param fromIndex the start of the range (inclusive), 0 by default.\n * **@param toIndex the end of the range (exclusive), size of this array by default.\n * **@throws IndexOutOfBoundsException if \[fromIndex\] is less than zero or \[toIndex\] is greater than the size of this array.\n * **@throws IllegalArgumentException if \[fromIndex\] is greater than \[toIndex\].\n * **@SinceKotlin("1.3")**

expect fun <T> Array<out T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Fills this array or its subrange with the specified \[element\] value.\n * **@param fromIndex the start of the range (inclusive), 0 by default.\n * **@param toIndex the end of the range (exclusive), size of this array by default.\n * **@throws IndexOutOfBoundsException if \[fromIndex\] is less than zero or \[toIndex\] is greater than the size of this array.\n * **@throws IllegalArgumentException if \[fromIndex\] is greater than \[toIndex\].\n * **@SinceKotlin("1.3")**

expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit

// **\n// Kotlin

```kotlin
public val ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
public expect fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit
```
indices for the array.

*\n
public val ShortArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val IntArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val FloatArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val DoubleArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val ByteArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val CharArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val BooleanArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val LongArray.indices: IntRange
    get() = IntRange(0, lastIndex)
    * Returns the range of valid indices for the array.

*\n
public val ShortArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val IntArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val FloatArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val DoubleArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val ByteArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val CharArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val BooleanArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val LongArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
public val ShortArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val IntArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val FloatArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val DoubleArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val ByteArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val CharArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val BooleanArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val LongArray.isNotEmpty(): Boolean
    get() = !isEmpty()
    * Returns `true` if the array is not empty.

*\n
public val ShortArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val IntArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val FloatArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val DoubleArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val ByteArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val CharArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val BooleanArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val LongArray.lastIndex: Int
    get() = size - 1
    * Returns the last valid index for the array.

*\n
public val ShortArray.plus(element: T): Array<T>
    get() = IntRange(0, lastIndex)
    * Returns an array containing all elements of the original array and then the given [element].

*\n
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect operator fun <T> Array<out T>.plus(element: T): Array<T>
    get() = IntRange(0, lastIndex)
    * Returns an array containing all elements of the original array and then the given [element].

*\n
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
@kotlin.internal.InlineOnly
public inline fun ByteArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
@kotlin.internal.InlineOnly
public inline fun ShortArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.

*\n
@kotlin.internal.InlineOnly
public inline fun IntArray.isEmpty(): Boolean
    get() = size == 0
    * Returns `true` if the array is empty.
CharArray.plus(elements: CharArray): CharArray

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NO_ACTUAL_FOR_EXPECT")
public fun <T> Array<T>.plusElement(element: T): Array<T> /* Sorts the array in-place. */

/**
 * Sorts the array in-place.
 */
public fun IntArray.sort(): IntArray

/**
 * Sorts the array in-place.
 */
public fun LongArray.sort(): LongArray

/**
 * Sorts the array in-place.
 */
public fun DoubleArray.sort(): DoubleArray

/**
 * Sorts the array in-place.
 */
public fun FloatArray.sort(): FloatArray

/**
 * Sorts the array in-place.
 */
public fun ByteArray.sort(): ByteArray

/**
 * Sorts the array in-place according to the natural order of its elements.
 */
public fun CharArray.sort(): CharArray

/**
 * Sorts the array in-place according to the order specified by the given [comparator].
 */
public fun <T : Comparable<T>> Array<out T>.sort(): Array<out T>

/**
 * Returns an array of Boolean containing all of the elements of this generic array.
 */
public fun ByteArray.toBooleanArray(): BooleanArray

/**
 * Returns an array of Byte containing all of the elements of this generic array.
 */
public fun ShortArray.toByteArray(): ByteArray

/**
 * Returns an array of Char containing all of the elements of this generic array.
 */
public fun IntArray.toCharArray(): CharArray

/**
 * Returns an array of Double containing all of the elements of this generic array.
 */
public fun LongArray.toDoubleArray(): DoubleArray

/**
 * Returns an array of Float containing all of the elements of this generic array.
 */
public fun FloatArray.toFloatArray(): FloatArray

/**
 * Returns an array of Int containing all of the elements of this generic array.
 */
public fun CharArray.toCharArray(): CharArray

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun ByteArray.toTypedArray(): Array<Byte>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun ShortArray.toTypedArray(): Array<Short>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun IntArray.toTypedArray(): Array<Int>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun LongArray.toTypedArray(): Array<Long>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun DoubleArray.toTypedArray(): Array<Double>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun FloatArray.toTypedArray(): Array<Float>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public fun BooleanArray.toTypedArray(): Array<Boolean>

/**
 * If any of two pairs would have the same key the last one gets added to the map.
 */
public fun Map<out Element, out Value> entrySet(): Set<Map.Entry<Element, Value>>
samples.collections.Arrays.Transformations.associateArrayOfPrimitives

/** public inline fun <T, K, V> 
 * Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <T, K, V> Array<out T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> ByteArray.associate(transform: (Byte) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> DoubleArray.associate(transform: (Double) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

/** Returns a [Map] containing key-value pairs provided by [transform] function
 * applied to elements of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}
samples.collections.Arrays.Transformations.associateArrayOfPrimitives

```kotlin
public inline fun <K, V> CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateTo(LinkedHashMap<K, V>(capacity), transform) }

public inline fun <K, V> CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateTo(LinkedHashMap<K, V>(capacity), transform) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <T, K> Array<out T>.associateBy(keySelector: (T) -> K): Map<K, T> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, T>(capacity), keySelector) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <K> ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Byte>(capacity), keySelector) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <K> ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Short>(capacity), keySelector) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <K> IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Int>(capacity), keySelector) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <K> LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Long>(capacity), keySelector) }

/**
 * Returns a [Map] containing the elements from the given array indexed by the key*
 * returned from [keySelector] function applied to each element.\n * \n * If any two elements would have the same key
 * returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration
 * order of the original array.\n * \n * @sample
 * samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy
 */
public inline fun <K> FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float> { val capacity = mapCapacity(size).coerceAtLeast(16) return associateByTo(LinkedHashMap<K, Float>(capacity), keySelector) }
```
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <T, V> Array<out T>.associateBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key
 * returned from [keySelector] function applied to each element.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>
have the same key returned by [keySelector] the last one gets added to the map.\n * \n * The returned map preserves the entry iteration order of the original array.\n * \n * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
val capacity = mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
val capacity = mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
val capacity = mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.\n * \n * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByWithValueTransform\n */
public inline fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
val capacity = mapCapacity(size).coerceAtLeast(16)
return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)\n}/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element of the given array\n * and value is the element itself.\n * \n * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo\n */
public inline fun <T, K, M : MutableMap<in K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
for (element in this) {
    destination.put(keySelector(element), element)\n} return destination\n}
two elements would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function applied to each element
 * of the given array * and value is the element itself. * If any two elements
 * would have the same key returned by [keySelector] the last one gets added to the map.

@sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByTo

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
map with key-value pairs,
where key is provided by the [keySelector] function and
value is provided by the [valueTransform] function applied to elements of the given array.
If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

@sample

```

```

@sample

```

```

@sample

```

```

@sample

```

```
destination.put(keySelector(element), valueTransform(element))

* Populates and returns the [destination] mutable map with key-value pairs.*

If any two elements would have the same key returned by [keySelector] the last one gets added to the map.

* @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueValueTransform: (Char) -> V): M {

* For (element in this) {

* destination.put(keySelector(element), valueTransform(element))

* return destination

* * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateTo(destination: M, transform: (Byte) -> Pair<K, V>): M {

* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {

* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {

* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo


* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {

* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key the last one gets added to the map.

* * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

* An public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M {

* For (element in this) {

* destination += transform(element)

* return destination

* * * Populates and returns the [destination] mutable map with key-value pairs

* provided by [transform] function applied to each element of the given array.

* If any of two pairs would have the same key
the last one gets added to the map.

```kotlin
// Open Source Used In DNAC Platform Arci

fun samples.collections.Arrays.Transformations.associateArrayOfPrimitivesToMap<K, V, M : MutableMap<K, in V>> BooleanArray.associateTo(destination: M, transform: (Boolean) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

fun samples.collections.Arrays.Transformations.associateArrayOfPrimitivesToMap<K, V, M : MutableMap<K, in V>> CharArray.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

fun <T, C : MutableCollection<in T>> Array<out T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Byte>> ByteArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Int>> IntArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Long>> LongArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Float>> FloatArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Double>> DoubleArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Boolean>> BooleanArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <C : MutableCollection<in Char>> CharArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

fun <T> Array<out T>.toList(): List<T> {
    return when (size) {
        0 -> emptyList()
        1 ->
Set<Long> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Long>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
*/
public fun FloatArray.toSet(): Set<Float> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Float>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
*/
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 * 
 * The returned set preserves the element iteration order of the original array.
*/
public fun BooleanArray.toSet(): Set<Boolean> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))
    }
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <T, R> Array<out T>.flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}
Open Source Used In DNAC Platform Arica  13059

transform: (T) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C : MutableCollection<R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
public inline fun <R, C :MutableCollection<R>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <T, K> Array<out T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> ByteArray.groupBy(keySelector: (Byte) -> K): Map<K, List<Byte>> {
    return groupByTo(LinkedHashMap<K, MutableList<Byte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> ShortArray.groupBy(keySelector: (Short) -> K): Map<K, List<Short>> {
    return groupByTo(LinkedHashMap<K, MutableList<Short>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> IntArray.groupBy(keySelector: (Int) -> K): Map<K, List<Int>> {
    return groupByTo(LinkedHashMap<K, MutableList<Int>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> LongArray.groupBy(keySelector: (Long) -> K): Map<K, List<Long>> {
    return groupByTo(LinkedHashMap<K, MutableList<Long>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> FloatArray.groupBy(keySelector: (Float) -> K): Map<K, List<Float>> {
    return groupByTo(LinkedHashMap<K, MutableList<Float>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> DoubleArray.groupBy(keySelector: (Double) -> K): Map<K, List<Double>> {
    return groupByTo(LinkedHashMap<K, MutableList<Double>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> BooleanArray.groupBy(keySelector: (Boolean) -> K): Map<K, List<Boolean>> {
    return groupByTo(LinkedHashMap<K, MutableList<Boolean>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of
corresponding elements.
 */
public inline fun <K> CharArray.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {
    return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)
}
original array by the key returned by the given [keySelector] function* applied to each element and returns a map where each group key is associated with a list of corresponding elements.* The returned map preserves the entry iteration order of the keys produced from the original array.* The returned map preserves the entry iteration order of the keys produced from the original array.* The returned map preserves the entry iteration order of the keys produced from the original array.* The returned map preserves the entry iteration order of the keys produced from the original array.*
the given [keySelector] function applied to the element\n * and returns a map where each group key is associated
with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */
public inline fun <K, V>
ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}

/**
 * Groups values
returned by the [valueTransform] function applied to each element of the original array in
 * by the key returned by
 * the given [keySelector] function applied to the element in
 * and returns a map where each group key is associated
 * with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */
public inline fun <K, V>
IntArray.groupBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}

/**
 * Groups values
returned by the [valueTransform] function applied to each element of the original array in
 * by the key returned by
 * the given [keySelector] function applied to the element in
 * and returns a map where each group key is associated
 * with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */
public inline fun <K, V>
LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}

/**
 * Groups values
returned by the [valueTransform] function applied to each element of the original array in
 * by the key returned by
 * the given [keySelector] function applied to the element in
 * and returns a map where each group key is associated
 * with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */
public inline fun <K, V>
FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}

/**
 * Groups values
returned by the [valueTransform] function applied to each element of the original array in
 * by the key returned by
 * the given [keySelector] function applied to the element in
 * and returns a map where each group key is associated
 * with a list of corresponding values.\n * \n * The returned map preserves the entry iteration order of the keys
produced from the original array.\n * \n * @sample
samples.collections.Collections.Transformations.groupByKeysAndValues\n */
public inline fun <K, V>
DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}

/**
 * Groups elements
of the original array by the key returned by the given [keySelector] function * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

```kotlin
// Inline function to group a list by a key selector.
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Array<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupByte function for grouping byte arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Byte>>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Byte>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupShort function for grouping short arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Short>>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Short>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupInt function for grouping integer arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Int>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Int>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupLong function for grouping long arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Long>>> LongArray.groupByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Long>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupFloat function for grouping float arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Float>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupDouble function for grouping double arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Double>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
// GroupBoolean function for grouping boolean arrays.
public inline fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<Boolean>() }
        list.add(element)
    }
    return destination
}
```
**Groups** elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.

```kotlin
destination\$[\n]mutableMap\$**\n* Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element and puts to the [destination] map each group key associated with a list of corresponding values.
```

```kotlin
destination\$[\n]mutableMap\$**\n* @return The [destination] map.
```

```kotlin
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(element)
    }
    return destination
}
```

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues\$[\n]*\n*public inline fun <T, K, V, M : MutableList<in K>, MutableList<in T>, MutableList<in V>, MutableList<in V>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues\$[\n]*\n*public inline fun <K, V, M : MutableList<in K>, MutableList<in T>, MutableList<in V>, MutableList<in V>> ByteArray.groupByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues\$[\n]*\n*public inline fun <K, V, M : MutableList<in K>, MutableList<in T>, MutableList<in V>, MutableList<in V>> ShortArray.groupByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues\$[\n]*\n*public inline fun <K, V, M : MutableList<in K>, MutableList<in T>, MutableList<in V>, MutableList<in V>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<in T>() }
        list.add(valueTransform(element))
    }
    return destination
}
```

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues\$[\n]*\n*public inline fun <K, V, M : MutableList<in K>, MutableList<in T>, MutableList<in V>, MutableList<in V>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M {
    for (element in this) {
        val key = keySelector(element)
```
val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableMap<K, V>, N> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {
    for (element in this)
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableMap<K, V>, N> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {
    for (element in this)
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the
 * original array by the key returned by the given [keySelector] function applied to the element
 * and puts to the [destination] map each group key associated with a list of corresponding values.
 * @return The [destination] map.
 * @sample samples.collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V, M : MutableMap<K, V>, N> CharArray.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this)
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from an array to be used later with one of group-and-fold
 * operations using the specified [keySelector] function to extract a key from each element.
 * @sample samples.collections.Grouping.groupingByEachCount
 */
@SinceKotlin("1.1")
public inline fun <T, K> Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun
            keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a list containing the results of applying
 * the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <T, R> Array<out T>.map(transform: (T) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
 * results of applying the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <R> ByteArray.map(transform: (Byte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
 * results of applying the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <R> ShortArray.map(transform: (Short) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
 * results of applying the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <R> IntArray.map(transform: (Int) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
 * results of applying the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <R> LongArray.map(transform: (Long) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the
 * results of applying the given [transform] function to each element in the original array.
 * @sample samples.collections.Transformations.map
 */
public inline fun <R> FloatArray.map(transform: (Float) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
transform applied to the element.

public inline fun <T, R : Any> Array<out T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends only the non-null results to the given [destination].
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <T, R, C : MutableCollection<in R>> Array<out T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> ShortArray.mapIndexedTo(destination: C, transform: (index: Int, Short) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> IntArray.mapIndexedTo(destination: C, transform: (index: Int, Int) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> LongArray.mapIndexedTo(destination: C, transform: (index: Int, Long) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> FloatArray.mapIndexedTo(destination: C, transform: (index: Int, Float) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> DoubleArray.mapIndexedTo(destination: C, transform: (index: Int, Double) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original array
 * and appends the results to the given [destination].
 */
public inline fun <R, C : MutableCollection<in R>> BooleanArray.mapIndexedTo(destination: C, transform: (index: Int, Boolean) -> R): C {
    var index = 0
    for (item in this)
        return destination.add(transform(index++, item))
    return destination
}
[transform] function to each element and its index in the original array\n * and appends the results to the given
[destination].\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
public inline fun <R, C : MutableCollection<in R>> CharArray.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element in the original array\n * and appends the results to the given [destination].\n */
public inline fun <T, R : Any> Array<out T>.mapNotNullTo(destination: C, transform: (T) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

/**
 * Applys the given [transform] function to each element of the original array\n * and appends the results to the given [destination].\n */
public inline fun <T, R : Any, C : MutableCollection<in R>> Array<out T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {
    for (element in this)
        destination.add(transform(element) ?: { destination.add(it) })
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}
index of that element and the element itself.

```kotlin
public fun IntArray.withIndex(): Iterable<IndexedValue<Int>> {
    return IndexingIterable { iterator() }
}

public fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}

public fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }
}

public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }
}

public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }
}

public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}

public fun <T> Array<out T>.distinct(): List<T> {
    return this.toMutableSet().toList()
}

public fun ByteArray.distinct(): List<Byte> {
    return this.toMutableSet().toList()
}

public fun ShortArray.distinct(): List<Short> {
    return this.toMutableSet().toList()
}

public fun IntArray.distinct(): List<Int> {
    return this.toMutableSet().toList()
}

public fun LongArray.distinct(): List<Long> {
    return this.toMutableSet().toList()
}

public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList()
}

public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList()
}

public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList()
}

public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList()
}
```

@sample
Open Source Used In DNAC Platform Arica  13069

```kotlin
/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations-distinctAndDistinctBy
public inline fun <T, K> Array<T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {
    val set = HashSet<K>()
    val list = ArrayList<Byte>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> ShortArray.distinctBy(selector: (Short) -> K): List<Short> {
    val set = HashSet<K>()
    val list = ArrayList<Short>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> IntArray.distinctBy(selector: (Int) -> K): List<Int> {
    val set = HashSet<K>()
    val list = ArrayList<Int>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> LongArray.distinctBy(selector: (Long) -> K): List<Long> {
    val set = HashSet<K>()
    val list = ArrayList<Long>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> FloatArray.distinctBy(selector: (Float) -> K): List<Float> {
    val set = HashSet<K>()
    val list = ArrayList<Float>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> DoubleArray.distinctBy(selector: (Double) -> K): List<Double> {
    val set = HashSet<K>()
    val list = ArrayList<Double>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> BooleanArray.distinctBy(selector: (Boolean) -> K): List<Boolean> {
    val set = HashSet<K>()
    val list = ArrayList<Boolean>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array
 * having distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they
 * were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy
public inline fun <K> CharArray.distinctBy(selector: (Char) -> K): List<Character> {
    val set = HashSet<K>()
    val list =
ArrayList<Char>() for (e in this) { 
    val key = selector(e)
    if (set.add(key))
        list.add(e)
}

return list

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* * An public infix fun <T> Array<out T>.intersect(other: Iterable<T>): Set<T> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun ByteArray.intersect(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun ShortArray.intersect(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun DoubleArray.intersect(other: Iterable<Double>): Set<Double> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun FloatArray.intersect(other: Iterable<Float>): Set<Float> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun IntArray.intersect(other: Iterable<Int>): Set<Int> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun LongArray.intersect(other: Iterable<Long>): Set<Long> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun CharArray.intersect(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun BooleanArray.intersect(other: Iterable<Boolean>): Set<Boolean> { 
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

* * An public infix fun CharArray.intersect(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ByteArray.subtract(other: Iterable<Byte>): Set<Byte> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}

* * An public infix fun ShortArray.subtract(other: Iterable<Short>): Set<Short> { 
    val set = this.toMutableSet()
    set.RemoveAll(other)
    return set
}
are contained by this array and not contained by the specified collection. The returned set preserves the

```kotlin
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
```

RM set contains all elements that are contained by this array and not contained by the specified collection.

```kotlin
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
```

Open Source Used In DNAC Platform Arica 13071
Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun <T> Array<out T> union(other: Iterable<T>): Set<T> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun ByteArray.union(other: Iterable<Byte>): Set<Byte> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun ShortArray.union(other: Iterable<Short>): Set<Short> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun DoubleArray.union(other: Iterable<Double>): Set<Double> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun FloatArray.union(other: Iterable<Float>): Set<Float> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun IntArray.union(other: Iterable<Int>): Set<Int> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun LongArray.union(other: Iterable<Long>): Set<Long> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun BooleanArray.union(other: Iterable<Boolean>): Set<Boolean> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns a set containing all distinct elements from both collections.\n* * * \n* The returned set preserves the element iteration order of the original array.\n* * \n* Those elements of the [other] collection that are unique are iterated in the end\n* in the order of the [other] collection.\n* * \n* To get a set containing all elements that are contained in both collections use [intersect].\n* * \n*public infix fun CharArray.union(other: Iterable<Char>): Set<Char> { 

| val set = this.toMutableSet()\n| set.addAll(other)\n| return set()\n|* */\n* * Returns `true` if all elements match the given [predicate].\n* * * \n* @sample \ncollections.Collections.Aggregates.all\n* *public inline fun <T> Array<out T> all(predicate: (T) -> Boolean): Boolean { 

| for (element in this) if (!predicate(element)) return false\n| return true\n|* */\n* * Returns `true` if all elements match the given [predicate].\n* * * \n* @sample \ncollections.Collections.Aggregates.all
public inline fun ShortArray.all(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun FloatArray.all(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ShortArray.anyWithPredicate(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun IntArray.anyWithPredicate(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun LongArray.anyWithPredicate(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun FloatArray.anyWithPredicate(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun DoubleArray.anyWithPredicate(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun BooleanArray.anyWithPredicate(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun CharArray.anyWithPredicate(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
samples.collections.Collections.Aggregates.anyWithPredicate

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun LongArray.anyWithPredicate(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun FloatArray.anyWithPredicate(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun DoubleArray.anyWithPredicate(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun BooleanArray.anyWithPredicate(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 */
public inline fun CharArray.anyWithPredicate(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.count(): Int {
    return size
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun <T> Array<out T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun IntArray.count(predicate: (Int) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and
 * applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <T, R>
Array<out T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.fold(initial: R, operation: (acc: R, Int) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> ShortArray.fold(initial: R, operation: (acc: R, Short) -> R): R {
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.

*\npublic inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this)
        accumulator = operation(index++, accumulator, element)
    return accumulator
} /* Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each element.

*\nThe accumulator is updated in place.
public fun <R> IntArray.foldRight(initial: R, operation: (Int, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index--, accumulator, value)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> ByteArray.foldRight(initial: R, operation: (Byte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index--, accumulator, element)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> ShortArray.foldRight(initial: R, operation: (Short, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index++, accumulator, element)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> LongArray.foldRight(initial: R, operation: (Long, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index++, accumulator, element)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> LongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Long) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (index: Int, acc: Boolean) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (index: Int, acc: Double) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to each element and current accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @return
 */
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
Open Source Used In DNAC Platform Arica 13077

```kotlin
var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * @param [initial] value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 */
public inline fun <R> FloatArray.foldRight(initial: R, operation: (Float, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * @param [initial] value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 */
public inline fun <R> DoubleArray.foldRight(initial: R, operation: (Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * @param [initial] value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 */
public inline fun <T, R> Array<out T>.foldRightIndexed(initial: R, operation: (index: Int, T, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * @param [initial] value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 */
public inline fun <R> CharArray.foldRight(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * @param [initial] value starting with [initial] value and applying [operation] from right to left
 * to each element with its index in the original array and current accumulator value.
 */
public inline fun <R> BooleanArray.foldRight(initial: R, operation: (Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Open Source Used In DNAC Platform Arica 13077
/**
 * Accumulates value starting with [initial] and current accumulator value.
 */
public inline fun <R> DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] and current accumulator value.
 */
public inline fun <R> BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] and current accumulator value.
 */
public inline fun <R> CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 */
public inline fun <T> Array<out T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun ByteArray.forEach(action: (Byte) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun ShortArray.forEach(action: (Short) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun IntArray.forEach(action: (Int) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun LongArray.forEach(action: (Long) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun FloatArray.forEach(action: (Float) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun DoubleArray.forEach(action: (Double) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun BooleanArray.forEach(action: (Boolean) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
public inline fun CharArray.forEach(action: (Char) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun <T> Array<out T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun BooleanArray.forEachIndexed(action: (index: Int, Boolean) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 */
public inline fun CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    ByteArray.maxBy(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    var lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    ShortArray.maxBy(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var maxElem = this[0]
    var lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    IntArray.maxBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var maxElem = this[0]
    var lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    LongArray.maxBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    FloatArray.maxBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    DoubleArray.maxBy(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    BooleanArray.maxBy(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
    CharArray.maxBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
Open Source Used In DNAC Platform Arica 13081
1..lastIndex) { 
    val e = this[i]
    if (min > e) min = e
} return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun IntArray.min(): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { 
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun LongArray.min(): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { 
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 */
public fun FloatArray.min(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    for (i in 1..lastIndex) { 
        val e = this[i]
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun DoubleArray.min(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { 
        val e = this[i]
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun CharArray.min(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { 
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { 
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> ByteArray.minBy(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { 
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> ShortArray.minBy(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { 
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { 
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { 
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var minElem = this[0]
    val
```kotlin
lastIndex = this.lastIndex
if (lastIndex == 0) return minElem
var minValue = selector(minElem)
for (i in 1..lastIndex) {
    val e = this[i]
    val v = selector(e)
    if (minValue > v) {
        minElem = e
        minValue = v
    }
}
return minElem
```

```kotlin
/**
 * Returns the first element yielding the smallest value of the given function or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minBy
 */
public inline fun <R : Comparable<R>>
    DoubleArray.minBy(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}
```

```kotlin
public fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minWith
 */
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minWith
 */
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minWith
 */
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or 'null' if there are no elements.
 * @sample samples.collections.Collections.Aggregates.minWith
 */
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
```
if (comparator.compare(min, e) > 0) min = e

public fun CharArray.minWith(comparator: Comparator<Char>): Char? {
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun <T> Array<T>.minWith(comparator: Comparator<T>): T? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the array has no elements.
 */
public fun <T> Array<T>.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun <T> Array<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun ByteArray.none(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun ShortArray.none(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun IntArray.none(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun LongArray.none(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun FloatArray.none(predicate: (Float) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun DoubleArray.none(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun BooleanArray.none(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample
 */
public inline fun CharArray.none(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (S, T : S) -> T): S {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Byte) -> Byte): Byte {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Short) -> Short): Short {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Int) -> Int): Int {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Long) -> Long): Long {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Float) -> Float): Float {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Double) -> Double): Double {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes
 (Char) -> Char): Char {
    for (element in this) if (predicate(element)) return false
    return true
}
the index of an element, current accumulator value\n * and the element itself and calculates the next accumulator
value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
public inline fun <S, T : S>
Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right\n * to current accumulator value and each element with its index in the original array.\n * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself and calculates the
next accumulator value.\n * in * @sample samples.collections.Collections.Aggregates.reduce\n */
DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
    if (isEmpty())
        throw
UnsupportedOperationException("Empty array can't be reduced.\n")
    var accumulator = this[0]\n    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return
accumulator\n
 * Accumulates value starting with the first element and applying [operation] from left to
right
* to current accumulator value and each element with its index in the original array.

* @param [operation] function that takes the index of an element, current accumulator value
* and the element itself and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.reduce

@public inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.
")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@public inline fun CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.
")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

@public inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    if (isEmpty())
        return null
    var accumulator: S = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

@public inline fun ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte): Byte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

@public inline fun ShortArray.reduceOrNull(operation: (acc: Short, Short) -> Short): Short? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

@public inline fun IntArray.reduceOrNull(operation: (acc: Int, Int) -> Int): Int? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

@public inline fun LongArray.reduceOrNull(operation: (acc: Long, Long) -> Long): Long? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

@public inline fun FloatArray.reduceOrNull(operation: (acc: Float, Float) -> Float): Float? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun DoubleArray.reduceOrNull(operation: (acc: Double, Double) -> Double): Double? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun BooleanArray.reduceOrNull(operation: (acc: Boolean, Boolean) -> Boolean): Boolean? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharArray.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.

```kotlin
public inline fun <S, T : S> Array<out T>.reduceRight(operation: (T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.

```kotlin
public inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.

```kotlin
public inline fun ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.

```kotlin
public inline fun IntArray.reduceRight(operation: (Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.

```kotlin
public inline fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```
samples.collections.Collections.Aggregates.reduceRight

```
/**
* Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.
* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceRight
*/
public inline fun FloatArray.reduceRightIndexed(operation: (Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```
/**
* Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.
* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceRight
*/
public inline fun DoubleArray.reduceRightIndexed(operation: (Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```
/**
* Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.
* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceRight
*/
public inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```
/**
* Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceRight
*/
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```
/**
* Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
* @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
* @sample samples.collections.Collections.Aggregates.reduceRight
*/
public inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```
IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int): Int 
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n** * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRight

    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n** * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRight

public inline fun FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float 
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n** * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRight

public inline fun DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double 
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n** * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRight

public inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean 
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n** * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRight

public inline fun CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char 
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRightOrNonNull

*\n@SinceKotlin("1.3") @ExperimentalStdlibApi \npublic inline fun LongArray.reduceRightOrNull(operation: (index: Int, Long) -> Long): Long 
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator

**Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.\n** * \n** * @sample samples.collections.Collections.Aggregates.reduceRightOrNonNull
accumulator) return accumulator
*/

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNullable
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun ByteArray.reduceRightOrNull(operation: (Byte, acc: Byte) -> Byte): Byte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun ShortArray.reduceRightOrNull(operation: (Short, acc: Short) -> Short): Short? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun IntArray.reduceRightOrNull(operation: (Int, acc: Int) -> Int): Int? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun LongArray.reduceRightOrNull(operation: (Long, acc: Long) -> Long): Long? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun FloatArray.reduceRightOrNull(operation: (Float, acc: Float) -> Float): Float? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun DoubleArray.reduceRightOrNull(operation: (Double, acc: Double) -> Double): Double? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun BooleanArray.reduceRightOrNull(operation: (Boolean, acc: Boolean) -> Boolean): Boolean? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun CharArray.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator =
get(index--) | while (index >= 0) {
accumulator = operation(get(index--), accumulator) 
} return accumulator

}\n\nReturns a list containing successive accumulation values generated by applying
[operation] from left to right in * to each element and current accumulator value that starts with [initial] value.

\n\n* Note that `acc` value passed to [operation] function should not be mutated;\n* otherwise it would affect the previous value in resulting list.

\n\n* @param [operation] function that takes current accumulator value and an
element, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.scan\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n\nopublic inline fun <T, R> Array<out T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> \n\nval result = ArrayList<R>(size + 1).apply { \n\nvar accumulator = initial\n\nfor (element in this) \n\naccumulator = operation(accumulator, element) \n\nresult.add(accumulator)\n\n} return result

\n\n* Returns a list containing successive accumulation values generated by applying
[operation] from left to right in * to each element and current accumulator value that starts with [initial] value.

\n\n* Note that `acc` value passed to [operation] function should not be mutated;\n* otherwise it would affect the previous value in resulting list.

\n\n* @param [operation] function that takes current accumulator value and an
element, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.scan

*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\n\nopublic inline fun <R>_byteArray.scan(initial: R, operation: (acc: R, Byte) -> R): List<R> \n\nval result = ArrayList<R>(size + 1).apply { \n\nvar accumulator = initial\n\nfor (element in this) \n\naccumulator = operation(accumulator, element) \n\nresult.add(accumulator)\n\n} return result

\n\n* Returns a list containing successive accumulation values generated by applying
[operation] from left to right in * to each element and current accumulator value that starts with [initial] value.

\n\n* Note that `acc` value passed to [operation] function should not be mutated;\n* otherwise it would affect the previous value in resulting list.

\n\n* @param [operation] function that takes current accumulator value and an
element, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.scan

*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\n\nopublic inline fun <R>_shortArray.scan(initial: R, operation: (acc: R, Short) -> R): List<R> \n\nval result = ArrayList<R>(size + 1).apply { \n\nvar accumulator = initial\n\nfor (element in this) \n\naccumulator = operation(accumulator, element) \n\nresult.add(accumulator)\n\n} return result

\n\n* Returns a list containing successive accumulation values generated by applying
[operation] from left to right in * to each element and current accumulator value that starts with [initial] value.

\n\n* Note that `acc` value passed to [operation] function should not be mutated;\n* otherwise it would affect the previous value in resulting list.

\n\n* @param [operation] function that takes current accumulator value and an
element, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.scan

*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\n\nopublic inline fun <R>_intArray.scan(initial: R, operation: (acc: R, Int) -> R): List<R> \n\nval result = ArrayList<R>(size + 1).apply { \n\nvar accumulator = initial\n\nfor (element in this) \n\naccumulator = operation(accumulator, element) \n\nresult.add(accumulator)\n\n} return result

\n\n* Returns a list containing successive accumulation values generated by applying
[operation] from left to right in * to each element and current accumulator value that starts with [initial] value.

\n\n* Note that `acc` value passed to [operation] function should not be mutated;\n* otherwise it would affect the previous value in resulting list.

\n\n* @param [operation] function that takes current accumulator value and an
element, and calculates the next accumulator value.

samples.collections.Collections.Aggregates.scan

*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\n\nopublic inline fun <R>_longArray.scan(initial: R, operation: (acc: R, Long) -> R): List<R> \n\nval result = ArrayList<R>(size + 1).apply { \n\nvar accumulator = initial\n\nfor (element in this) \n\naccumulator = operation(accumulator, element) \n\nresult.add(accumulator)\n\n} return result

\n\n* Returns a list containing successive accumulation values generated by applying
[operation] from left to right in * to
each element and current accumulator value that starts with [initial] value.

Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

@sample samples.collections.Collections.Aggregates.scan

/*An@SinceKotlin("1.3")@n@ExperimentalStdlibApi@n@kotlin.internal.InlineOnly@n@public inline fun <R>
FloatArray.scan(initial: R, operation: (acc: R, Float) -> R): List<R>
{ val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        add(accumulator)
    }
    return result
}*/

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
DoubleArray.scan(initial: R, operation: (acc: R, Double) -> R): List<R>
{ val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        add(accumulator)
    }
    return result
}

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
BooleanArray.scan(initial: R, operation: (acc: R, Boolean) -> R): List<R>
{ val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        add(accumulator)
    }
    return result
}

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R>
{ val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        add(accumulator)
    }
    return result
}

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <T, R> Array<out T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R>
{ val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        add(accumulator)
    }
    return result
}
DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

BooleanArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

ByteArray.scanReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size + 1).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

CharArray.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size + 1).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

ByteArray.scanReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size + 1).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

CharArray.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(size + 1).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

```kotlin
*\n* @sample samples.collections.Collections.Aggregates.scanReduce

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun ShortArray.scanReduce(operation: (acc: Short, Short) -> Short): List<Short> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun IntArray.scanReduce(operation: (acc: Int, Int) -> Int): List<Int> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun LongArray.scanReduce(operation: (acc: Long, Long) -> Long): List<Long> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Long>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun FloatArray.scanReduce(operation: (acc: Float, Float) -> Float): List<Float> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Float>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun DoubleArray.scanReduce(operation: (acc: Double, Double) -> Double): List<Double> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Double>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@public inline fun BooleanArray.scanReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Boolean>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right \* to each element and current accumulator value that starts with the first element of this array.
starts with the first element of this array.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scanReduce

/*@SinceKotlin("1.3")@*/@ExperimentalStdlibApi@public inline fun CharArray.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    var accumulator = this[0]
    val result = ArrayList<Chars>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated.
 * otherwise it would affect the previous value in resulting list.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scanReduce

/*@SinceKotlin("1.3")@*/@ExperimentalStdlibApi@public inline fun StringArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    var accumulator = this[0]
    val result = ArrayList<Chars>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scanReduce

/*@SinceKotlin("1.3")@*/@ExperimentalStdlibApi@public inline fun ByteArray.scanReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {
    var accumulator = this[0]
    val result = ArrayList<Chars>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scanReduce

/*@SinceKotlin("1.3")@*/@ExperimentalStdlibApi@public inline fun ShortArray.scanReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {
    var accumulator = this[0]
    val result = ArrayList<Chars>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array.

* @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scanReduce

/*@SinceKotlin("1.3")@*/@ExperimentalStdlibApi@public inline fun IntArray.scanReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {
    var accumulator = this[0]
    val result = ArrayList<Chars>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
accumulator.value

*\n
*\n
@sample samples.collections.Collections.Aggregates.scanReduce

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.scanReduceIndexed(operation: (index: Int, acc: Long, Long) -> Long): List<Long> {\n    var accumulator = this[0]\n    val result = ArrayList<Long>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        return result.add(accumulator)\n    }\n    return emptyList()\n}

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun FloatArray.scanReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {\n    var accumulator = this[0]\n    val result = ArrayList<Float>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        return result.add(accumulator)\n    }\n    return emptyList()\n}

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun DoubleArray.scanReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {\n    var accumulator = this[0]\n    val result = ArrayList<Double>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        return result.add(accumulator)\n    }\n    return emptyList()\n}

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun BooleanArray.scanReduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): List<Boolean> {\n    var accumulator = this[0]\n    val result = ArrayList<Boolean>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        return result.add(accumulator)\n    }\n    return emptyList()\n}

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {\n    var accumulator = this[0]\n    val result = ArrayList<Char>(size).apply { add(accumulator) }\n    for (index in 1 until size) {\n        accumulator = operation(index, accumulator, this[index])\n        return result.add(accumulator)\n    }\n    return emptyList()\n}

*\n
*\n
@sample samples.collections.Collections.Aggregates.scanReduce

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum
}

*\n
*\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun Array<out T>.sumBy(selector: (T) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumBy(selector: (Byte) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumBy(selector: (Short) -> Int): Int {\n    var sum: Int = 0\n    for (element in this) {\n        sum += selector(element)\n    }\n    return sum
}
```java
var sum: Int = 0
for (element in this) {
    sum += selector(element)
}
return sum

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumBy(selector: (Int) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumBy(selector: (Long) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumBy(selector: (Float) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumBy(selector: (Double) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumBy(selector: (Boolean) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ByteArray.sumByDouble(selector: (Byte) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ShortArray.sumByDouble(selector: (Short) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun IntArray.sumByDouble(selector: (Int) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun LongArray.sumByDouble(selector: (Long) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun FloatArray.sumByDouble(selector: (Float) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun DoubleArray.sumByDouble(selector: (Double) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun BooleanArray.sumByDouble(selector: (Boolean) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun CharArray.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns an original collection containing all the non-null elements, throwing an [IllegalArgumentExce
```
T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
  val first = ArrayList<T>(element for (element in this) if (predicate(element))
  val second = ArrayList<T>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

ByteArray.partition(predicate: (Byte) -> Boolean): Pair<List<Byte>, List<Byte>> {
  val first = ArrayList<Byte>(element for (element in this) if (predicate(element))
  val second = ArrayList<Byte>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

ShortArray.partition(predicate: (Short) -> Boolean): Pair<List<Short>, List<Short>> {
  val first = ArrayList<Short>(element for (element in this) if (predicate(element))
  val second = ArrayList<Short>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

IntArray.partition(predicate: (Int) -> Boolean): Pair<List<Int>, List<Int>> {
  val first = ArrayList<Int>(element for (element in this) if (predicate(element))
  val second = ArrayList<Int>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

LongArray.partition(predicate: (Long) -> Boolean): Pair<List<Long>, List<Long>> {
  val first = ArrayList<Long>(element for (element in this) if (predicate(element))
  val second = ArrayList<Long>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

FloatArray.partition(predicate: (Float) -> Boolean): Pair<List<Float>, List<Float>> {
  val first = ArrayList<Float>(element for (element in this) if (predicate(element))
  val second = ArrayList<Float>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

DoubleArray.partition(predicate: (Double) -> Boolean): Pair<List<Double>, List<Double>> {
  val first = ArrayList<Double>(element for (element in this) if (predicate(element))
  val second = ArrayList<Double>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/

BooleanArray.partition(predicate: (Boolean) -> Boolean): Pair<List<Boolean>, List<Boolean>> {
  val first = ArrayList<Boolean>(element for (element in this) if (predicate(element))
  val second = ArrayList<Boolean>(element for (element in this) else
  return Pair(first, second)/* Splits the original array into pair of lists
  * where *first* list contains elements for which [predicate] yielded true;
  * while *second* list contains elements for which [predicate] yielded false.*/
}/* @sample samples.collections.Arrays.Transformations.partitionArrayOfPrimitives*/
samples.collections.Iterables.Operations.zipIterableWithTransform
*/

public inline fun <R, V> ShortArray.zip(other: Array<out R>, transform: (a: Short, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> IntArray.zip(other: Array<out R>, transform: (a: Int, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> LongArray.zip(other: Array<out R>, transform: (a: Long, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> FloatArray.zip(other: Array<out R>, transform: (a: Float, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> DoubleArray.zip(other: Array<out R>, transform: (a: Double, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {
    val size = minOf(this.size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <T, R> Array<out T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R, V> ByteArray.zip(other: Iterable<R>): List<Pair<Byte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of this array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. 
 * The returned list has length of the shortest collection.
 */
@sample samples.collections.Iterables.Operations.zipIterableWithTransform

public inline fun <R> ShortArray.zip(other: Iterable<R>): List<Pair<Short, R>> { return zip(other) { t1, t2 -> t1 to t2 } }
List<Pair<Short, R>> {\n    \n    return zip(other) { t1, t2 -> t1 to t2 }\n    \n    * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n    * The returned list has length of the shortest collection.\n    * \n    }\n    * @sample samples.collections.Operations.zip\n\n    intArray.zip(other: Iterable<Int>): List<Pair<Int, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> IntArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> LongArray.zip(other: Iterable<R>): List<Pair<Long, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public infix fun <R> CharArray.zip(other: Iterable<R>): List<Pair<Char, R>> {\n        \n        return zip(other) { t1, t2 -> t1 to t2 }\n        \n        * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zip\n\n    public <T, R, V> Array<out T>.zip(other: Iterable<R>, transform: (a: T, b: R) -> V): List<V> {\n        \n        val arraySize = size\n        var list = ArrayList<V>(\n            minOf(other.collectionSizeOrDefault(10), arraySize))\n        \n        for (element in other) {\n            \n            list.add(transform(this[i++], element))\n            \n            return list\n        }\n        \n        * Returns a list of values built from the elements of `this` array and the [other] collection with the provided [transform] function applied to each pair of elements.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zipWithTransform\n\n    public <R, V> ByteArray.zip(other: Iterable<R>, transform: (a: Byte, b: R) -> V): List<V> {\n        \n        val arraySize = size\n        var list = ArrayList<V>(\n            minOf(other.collectionSizeOrDefault(10), arraySize))\n        \n        for (element in other) {\n            \n            list.add(transform(this[i++], element))\n            \n            return list\n        }\n        \n        * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n        * using the provided [transform] function applied to each pair of elements.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zipWithTransform\n\n    public <R, V> ShortArray.zip(other: Iterable<R>, transform: (a: Short, b: R) -> V): List<V> {\n        \n        val arraySize = size\n        var list = ArrayList<V>(\n            minOf(other.collectionSizeOrDefault(10), arraySize))\n        \n        for (element in other) {\n            \n            list.add(transform(this[i++], element))\n            \n            return list\n        }\n        \n        * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n        * using the provided [transform] function applied to each pair of elements.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zipWithTransform\n\n    public <R, V> IntArray.zip(other: Iterable<R>, transform: (a: Int, b: R) -> V): List<V> {\n        \n        val arraySize = size\n        var list = ArrayList<V>(\n            minOf(other.collectionSizeOrDefault(10), arraySize))\n        \n        for (element in other) {\n            \n            list.add(transform(this[i++], element))\n            \n            return list\n        }\n        \n        * Returns a list of values built from the elements of `this` array and the [other] collection with the same index\n        * using the provided [transform] function applied to each pair of elements.\n        * The returned list has length of the shortest collection.\n        * \n    }\n    * @sample samples.collections.Operations.zipWithTransform\n\n    Open Source Used in DNAC Platform Arica 13103
provided [transform] function applied to each pair of elements.

The returned list has length of the shortest collection.

@sample samples.collections.Operations.zipIterableWithTransform

    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

fun <R, V> FloatArray.zip(other: Iterable<R>, transform: (a: Float, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

fun <R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

fun ByteArray.zip(other: ByteArray): List<Pair<Byte, Byte>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun ShortArray.zip(other: ShortArray): List<Pair<Short, Short>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun IntArray.zip(other: IntArray): List<Pair<Int, Int>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun LongArray.zip(other: LongArray): List<Pair<Long, Long>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

fun FloatArray.zip(other: FloatArray):
List<Pair<Float, Float>> { return zip(other) { t1, t2 -> t1 to t2 } }

The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable

** @sample

DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> { return zip(other) { t1, t2 -> t1 to t2 } }

The returned list has length of the shortest collection.

* @sample samples.collections.Iterables.Operations.zipIterable

CharArray.zip(other: CharArray): List<Pair<Char, Char>> { return zip(other) { t1, t2 -> t1 to t2 } }

* @sample samples.collections.Iterables.Operations.zipIterable

ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Iterables.Operations.zipIterable

ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Iterables.Operations.zipIterable

IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Iterables.Operations.zipIterable

LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Iterables.Operations.zipIterable

FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samplescollections.Operations.zipIterable

DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.Operations.zipIterable

Open Source Used In DNAC Platform Arica  13105

List<Pair<Float, Float>> { return zip(other) { t1, t2 -> t1 to t2 } }

The returned list has length of the shortest collection.

* @sample samples.collections.Operations.zipIterable

** @sample

DoubleArray.zip(other: DoubleArray): List<Pair<Double, Double>> { return zip(other) { t1, t2 -> t1 to t2 } }

The returned list has length of the shortest collection.

* @sample samples.collections.Operations.zipIterable

CharArray.zip(other: CharArray): List<Pair<Char, Char>> { return zip(other) { t1, t2 -> t1 to t2 } }

* @sample samples.collections.Operations.zipIterable

ByteArray.zip(other: ByteArray, transform: (a: Byte, b: Byte) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Operations.zipIterable

ShortArray.zip(other: ShortArray, transform: (a: Short, b: Short) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Operations.zipIterable

IntArray.zip(other: IntArray, transform: (a: Int, b: Int) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Operations.zipIterable

LongArray.zip(other: LongArray, transform: (a: Long, b: Long) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Operations.zipIterable

FloatArray.zip(other: FloatArray, transform: (a: Float, b: Float) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.collections.Operations.zipIterable

DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: Double) -> V): List<V> { val size = minOf(size, other.size) val list = ArrayList<V>(size) for (i in 0 until size) { list.add(transform(this[i], other[i])) } return list }

* @sample samples.Operations.zipIterable

Open Source Used In DNAC Platform Arica  13105
samples.collections.Iterables.Operations.zipIterableWithTransform\n/*\n\npublic inline fun <V>

BooleanArray.zip(other: BooleanArray, transform: (a: Boolean, b: Boolean) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList\n        \n    for (i in 0 until size) \n        list.add\n        \n    return list\n\n\n    * Returns a list of values built from the elements of ‘this’ array and the \n    * array with the same index\n    * using the provided [transform] function applied to each pair of elements.\n\n    The returned list has length of the shortest array.\n\n    \n\n@sample \n\n\nOpen Source Used In DNAC Platform Arica  13106

samples.collections.Iterables.Operations.zipIterableWithTransform\n/*\n\npublic inline fun <V>

CharArray.zip(other: CharArray, transform: (a: Char, b: Char) -> V): List<V> \n{\n    val size = minOf(size, other.size)\n    val list = ArrayList\n        \n    for (i in 0 until size) \n        list.add\n        \n    return list\n\n\n    * Returns a list of values built from the elements of ‘this’ array and the \n    * array with the same index\n    * using the provided [transform] function applied to each pair of elements.\n\n    The returned list has length of the shortest array.\n\n    \n\n@sample \n\n\nOpen Source Used In DNAC Platform Arica  13106
/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. \n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."). \n * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * @sample samples.collections.Collections.Transformations.joinTo
 */

public fun <A : Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. \n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."). \n * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * @sample samples.collections.Collections.Transformations.joinTo
 */

public fun <A : Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. \n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."). \n * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * @sample samples.collections.Collections.Transformations.joinTo
 */

public fun <A : Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. \n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."). \n * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * @sample samples.collections.Collections.Transformations.joinTo
 */

public fun <A : Appendable> BooleanArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Appends the string from all the elements separated using [separator] and using the given
 * [prefix] and [postfix] if supplied. \n * \n * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "..."). \n * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * @sample samples.collections.Collections.Transformations.joinTo
 */

public fun <A : Appendable> CharArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(separator)
        if (limit < 0 || count <= limit) {
            if (transform != null)
                buffer.append(transform(element))
            else
                buffer.append(element.toString())
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}
buffer.append(transform(element))
else
    buffer.append(element)
}

} else break
  }
    if

    if

    buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 *
* Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
* If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
* @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Array<out T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun ByteArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Byte) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun ShortArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Short) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun IntArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun LongArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun FloatArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun DoubleArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
CharSequence = ",", prefix: CharSequence = ",", postfix: CharSequence = ",", limit: Int = -1, truncated:
CharSequence = ",...", transform: ((Double) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to ",...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun BooleanArray.joinToString(separator: CharSequence = ",", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to ",...").
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun CharArray.joinToString(separator: CharSequence = ",", prefix: CharSequence = "", postfix: CharSequence = ",", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun <T> Array<out T>.asIterable(): Iterable<T> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun ByteArray.asIterable(): Sequence<Byte> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun ShortArray.asIterable(): Sequence<Short> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun IntArray.asIterable(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun LongArray.asIterable(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun FloatArray.asIterable(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun DoubleArray.asIterable(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun ByteArray.asSequence(): Sequence<Byte> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun ShortArray.asSequence(): Sequence<Short> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun IntArray.asSequence(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun FloatArray.asSequence(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 * @sample samples.collections.Sequences.Building.sequenceFromArray
 */
public fun DoubleArray.asSequence(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
Sequence<Int> { 
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
@sample samples.collections.Sequences.Building.sequenceFromArray

public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
@sample samples.collections.Sequences.Building.sequenceFromArray

public fun FloatArray.asSequence(): Sequence<Float> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
@sample samples.collections.Sequences.Building.sequenceFromArray

public fun DoubleArray.asSequence(): Sequence<Double> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
@sample samples.collections.Sequences.Building.sequenceFromArray

public fun BooleanArray.asSequence(): Sequence<Boolean> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */
@sample samples.collections.Sequences.Building.sequenceFromArray

public fun CharArray.asSequence(): Sequence<Char> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Array<out Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Array<out Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Array<out Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Array<out Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Array<out Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Array<out Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ByteArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun ShortArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun IntArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun LongArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the array.
 */
public fun FloatArray.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}
DoubleArray.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        ++count
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Array<out Byte>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Array<out Short>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Array<out Int>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Array<out Long>.sum(): Long {
    var sum = Long(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Array<out Float>.sum(): Float {
    var sum = Float(0.0f)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Array<out Double>.sum(): Double {
    var sum = Double(0.0)
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming
 * Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    return random.nextInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun LongRange.random(random: Random): Long {
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun CharRange.random(): Char {
    return random.nextChar()
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun LongRange.random(random: Random): Long {
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun CharRange.random(): Char {
    return random.nextChar()
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    return random.nextInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun LongRange.random(random: Random): Long {
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun CharRange.random(): Char {
    return random.nextChar()
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun IntRange.random(random: Random): Int {
    return random.nextInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun LongRange.random(random: Random): Long {
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
public fun CharRange.random(): Char {
    return random.nextChar()
public fun CharRange.random(random: Random): Char {
    try {
        return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
    } catch (e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
public inline fun IntRange.randomOrNull(): Int? {
    if (isEmpty())
        return null
    return random.nextInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public fun IntRange.randomOrNull(random: Random): Int? {
    if (isEmpty())
        return null
    return random.nextInt(this)
}

/**
 * Returns a random element from this range, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public fun LongRange.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return null
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public fun LongRange.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return null
    return random.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public fun CharRange.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Returns a random element from this range using the specified source of randomness, or ‘null’ if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public fun CharRange.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Returns `true` if this range contains the specified [element].
 * 
 * Always returns ‘false’ if the [element] is ‘null’.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public operator fun IntRange.contains(value: Byte): Boolean {
    return contains(value.toInt())
}

/**
 * Returns `true` if this range contains the specified [element].
 * 
 * Always returns ‘false’ if the [element] is ‘null’.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public operator fun LongRange.contains(value: Byte): Boolean {
    return contains(value.toLong())
}

/**
 * Returns `true` if this range contains the specified [element].
 * 
 * Always returns ‘false’ if the [element] is ‘null’.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public operator fun CharRange.contains(value: Byte): Boolean {
    return contains(value.toChar())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Double): Boolean {
    return value.toIntExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Double): Boolean {
    return value.toLongOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Double): Boolean {
    return value.toShortOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Byte): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Byte): Boolean {
    return contains(value.toFloat())
}
and floating point arguments has ambiguous semantics and is going to be removed.

```kotlin
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Double): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@deprecated("This contains operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Double): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@deprecated("This contains operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Double): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Double): Boolean {
    return contains(value.toDouble())
}
```

Open Source Used In DNAC Platform Arica 13113
ClosedRange<Int>.contains(value: Long): Boolean {
    return value.toIntOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Long): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Long): Boolean {
    return value.toShortExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Long): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Long): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Short): Boolean {
    return contains(value.toLong())
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Int.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Long.downTo(to: Byte): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Byte.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Short.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Char.downTo(to: Char): CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * 
 * The [to] value should be less than or equal to `this` value. If the [to] value is greater than `this` value the returned progression is empty.
 */
public infix fun Int.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this,
to, -1)n

\n\n*/

* Returns a progression from this value down to the specified [to] value with the step -1.n

\n* The [to] value should be less than or equal to `this` value.n

* If the [to] value is greater than `this` value the returned progression is empty.n

*\n
*/

\n
*public infix fun Long.downTo(to: Int): LongProgression {

* return

LongProgression.fromClosedRange(this, to.toLong(), -1L)n\n\n*/

* Returns a progression from this value down to the specified [to] value with the step -1.n

\n* The [to] value should be less than or equal to `this` value.n

* If the [to] value is greater than `this` value the returned progression is empty.n

*\n
*/

\n
*public infix fun Byte.downTo(to: Int): IntProgression {

* return

IntProgression.fromClosedRange(this.toInt(), to, -1)n\n\n*/

* Returns a progression from this value down to the specified [to] value with the step -1.n

\n* The [to] value should be less than or equal to `this` value.n

* If the [to] value is greater than `this` value the returned progression is empty.n

*\n
*/

\n
*public infix fun Short.downTo(to: Int): IntProgression {

* return

IntProgression.fromClosedRange(this.toInt(), to, -1)n\n\n*/

* Returns a progression from this value down to the specified [to] value with the step -1.n

\n* The [to] value should be less than or equal to `this` value.n

* If the [to] value is greater than `this` value the returned progression is empty.n

*\n
*/

\n
*public infix fun Int.downTo(to: Int): IntProgression {

* return

IntProgression.fromClosedRange(this, to.toInt(), -1)n\n\n*/

* Returns a progression from this value down to the specified [to] value with the step -1.n

\n* The [to] value should be less than or equal to `this` value.n

* If the [to] value is greater than `this` value the returned progression is empty.n

*\n
*/

\n
*public infix fun IntProgression.reversed(): IntProgression {

* return

IntProgression.fromClosedRange(last, first, -step)n\n\n*/

* Returns a progression that goes over the same range in the opposite direction with the same step.

*/

*public infix fun IntProgression.step(step: Int): IntProgression {

* checkStepIsPositive(step > 0, step)n

* return

IntProgression.fromClosedRange(last, first, -step)n\n\n*/

* Returns a progression that goes over the same range in the opposite direction with the same step.

*/

*public infix fun IntProgression.reversed(): IntProgression {

* return

IntProgression.fromClosedRange(last, first, -step)n\n\n*/
LongProgression.step(step: Long): LongProgression {
    checkStepIsPositive(step > 0, step)
    return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
public infix fun CharProgression.step(step: Int): CharProgression {
    checkStepIsPositive(step > 0, step)
    return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

internal fun Int.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toInt()..Byte.MAX_VALUE.toInt()) this.toByte() else null
}

internal fun Long.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toLong()..Byte.MAX_VALUE.toLong()) this.toByte() else null
}

internal fun Short.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toShort()..Byte.MAX_VALUE.toShort()) this.toByte() else null
}

internal fun Double.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toByte() else null
}

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toByte() else null
}

internal fun Int.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun Float.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toFloat()..Int.MAX_VALUE.toFloat()) this.toInt() else null
}

internal fun Double.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toDouble()..Long.MAX_VALUE.toDouble()) this.toLong() else null
}

internal fun Float.toLongExactOrNull(): Long? {
    return if (this in Long.MIN_VALUE.toFloat()..Long.MAX_VALUE.toFloat()) this.toLong() else null
}

.internal fun Short.toLongExactOrNull(): Long? {
    return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt()) this.toLong() else null
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Int.until(to: Byte): IntRange {
    return this .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Long.until(to: Byte): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Byte.until(to: Byte): IntRange {
    return this.toInt() .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Char.until(to: Char): CharRange {
    if (to <= Char.MIN_VALUE) return CharRange.EMPTY
    return this .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Int.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this .. (totoInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Long.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Byte.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Char.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Int.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this .. (totoInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Long.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Byte.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Char.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Int.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this .. (totoInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Long.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Byte.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Char.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Int.until(to: Int): IntRange {
    if (to <= Int.MIN_VALUE) return IntRange.EMPTY
    return this .. (totoInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Long.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Byte.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
public infix fun Char.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
If the [to] value is less than or equal to `this` value, then the returned range is empty.

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Int): IntRange {
    return if (to <= Int.MIN_VALUE) IntRange.EMPTY
              else this.toInt() .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Int): IntRange {
    return if (to <= Int.MIN_VALUE) IntRange.EMPTY
              else this.toInt() .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Long): LongRange {
    return if (to <= Long.MIN_VALUE) LongRange.EMPTY
              else this.toLong() .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Long): LongRange {
    return if (to <= Long.MIN_VALUE) LongRange.EMPTY
              else this.toLong() .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Long): LongRange {
    return if (to <= Long.MIN_VALUE) LongRange.EMPTY
              else this.toLong() .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Long): LongRange {
    return if (to <= Long.MIN_VALUE) LongRange.EMPTY
              else this.toLong() .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Byte.coerceAtLeast(minimumValue: Byte): Byte {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Short.coerceAtLeast(minimumValue: Short): Short {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Int.coerceAtLeast(minimumValue: Int): Int {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Long.coerceAtLeast(minimumValue: Long): Long {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Double.coerceAtLeast(minimumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun String.coerceAtLeast(minimumValue: String): String {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * @return this value if it's greater than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Any.coerceAtLeast(minimumValue: Any): Any {
    return if (this < minimumValue) minimumValue else this
}
```
public fun Float.coerceAtLeast(minimumValue: Float): Float {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [minimumValue].
 * @return this value if it's less than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Double.coerceAtLeast(minimumValue: Double): Double {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [minimumValue].
 * @return this value if it's less than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Int.coerceAtLeast(minimumValue: Int): Int {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [minimumValue].
 * @return this value if it's less than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Long.coerceAtLeast(minimumValue: Long): Long {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not greater than the specified [minimumValue].
 * @return this value if it's less than or equal to the [minimumValue] or the [minimumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtLeast
 */
public fun Float.coerceAtMost(maximumValue: Float): Float {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Double.coerceAtMost(maximumValue: Double): Double {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Float.coerceAtMost(maximumValue: Float): Float {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMost
 */
public fun Double.coerceAtMost(maximumValue: Double): Double {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {
    return if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    else if (this < minimumValue) minimumValue
    else if (this > maximumValue) maximumValue
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {
    return if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    else if (this < minimumValue) minimumValue
    else if (this > maximumValue) maximumValue
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {
    return if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    else if (this < minimumValue) minimumValue
    else if (this > maximumValue) maximumValue
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {
    return if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    else if (this < minimumValue) minimumValue
    else if (this > maximumValue) maximumValue
}
return maximumValue\n    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or minimumValue if it's less than minimumValue, or maximumValue if it's greater than maximumValue.
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Short.coerceIn(minimumValue: Short, maximumValue: Short): Short {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or minimumValue if it's less than minimumValue, or maximumValue if it's greater than maximumValue.
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Int.coerceIn(minimumValue: Int, maximumValue: Int): Int {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or minimumValue if it's less than minimumValue, or maximumValue if it's greater than maximumValue.
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Long.coerceIn(minimumValue: Long, maximumValue: Long): Long {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or minimumValue if it's less than minimumValue, or maximumValue if it's greater than maximumValue.
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun Float.coerceIn(minimumValue: Float, maximumValue: Float): Float {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * @sample samples.comparisons.ComparableOps.coerceInFloatingPointRange
 */
@SinceKotlin("1.1")
public fun <T : Comparable<T>> T.coerceIn(range: ClosedFloatingPointRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue..maximumValue].
 * @return this value if it's in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * @sample samples.comparisons.ComparableOps.coerceInComparable
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return
    }
}
other value from this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun minus(other: UShort):
  UInt = this.toUInt().minus(other.toUInt())
  /** Subtracts the other value from this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun minus(other: UInt):
  UInt = this.toUInt().minus(other)
  /** Subtracts the other value from this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun minus(other: ULong):
  ULong = this.toULong().minus(other)
  /** Subtracts the other value from this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun minus(other: ULong):
  ULong = this.toULong().minus(other)

  /** Multiplies this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun times(other: UByte):
  UInt = this.toUInt().times(other.toUInt())
  /** Multiplies this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun times(other: UShort):
  UInt = this.toUInt().times(other.toUInt())
  /** Multiplies this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun times(other: UInt):
  UInt = this.toUInt().times(other)
  /** Multiplies this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun times(other: ULong):
  ULong = this.toULong().times(other)

  /** Divides this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun div(other: UByte):
  UInt = this.toUInt().div(other.toUInt())
  /** Divides this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun div(other: UShort):
  UInt = this.toUInt().div(other.toUInt())
  /** Divides this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun div(other: UInt):
  UInt = this.toUInt().div(other)
  /** Divides this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun div(other: ULong):
  ULong = this.toULong().div(other)

  /** Calculates the remainder of dividing this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun rem(other: UByte):
  UInt = this.toUInt().rem(other.toUInt())
  /** Calculates the remainder of dividing this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun rem(other: UShort):
  UInt = this.toUInt().rem(other.toUInt())
  /** Calculates the remainder of dividing this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun rem(other: UInt):
  UInt = this.toUInt().rem(other)
  /** Calculates the remainder of dividing this value by the other value. */
  @kotlin.internal.InlineOnly
  public inline operator fun rem(other: ULong):
  ULong = this.toULong().rem(other)

  /** Increments this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun inc(): UByte = UByte(data.inc())
  /** Decrements this value. */
  @kotlin.internal.InlineOnly
  public inline operator fun dec(): UByte = UByte(data.dec())

  /** Creates a range from this value to the specified [other] value. */
  @kotlin.internal.InlineOnly
  public inline operator fun rangeTo(other: UByte):
  UIntRange = UIntRange(this.toUInt(), other.toUInt())

  /** Performs a bitwise AND operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline infix fun and(other: UByte): UByte = UByte(this.data and other.data)
  /** Performs a bitwise OR operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline infix fun or(other: UByte): UByte = UByte(this.data
or other.data)
  /** Performs a bitwise XOR operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline infix fun xor(other: UByte): UByte = UByte(this.data
xor other.data)
  /** Inverts the bits in this value. */
  @kotlin.internal.InlineOnly
  public inline fun inv(): UByte = UByte(data.inv())

  /** Converts this [UByte] value to [Byte]. */
  @kotlin.internal.InlineOnly
  public inline fun toByte(): Byte = data

  /** Converts this [UByte] value to [Short]. */
  @kotlin.internal.InlineOnly
  public inline fun toShort(): Short = data.toShort()

  /** Converts this [UByte] value to [Int]. */
  @kotlin.internal.InlineOnly
  public inline fun toInt(): Int = data.toInt()

  /** Converts this [UByte] value to [Long]. */
  @kotlin.internal.InlineOnly
  public inline fun toLong(): Long = data.toLong()

  /** Performs a bitwise AND operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline fun and(other: UByte): UByte =
  UByte(this.data and other.data)

  /** Performs a bitwise OR operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline fun or(other: UByte): UByte =
  UByte(this.data or other.data)

  /** Performs a bitwise XOR operation between the two values. */
  @kotlin.internal.InlineOnly
  public inline fun xor(other: UByte): UByte =
  UByte(this.data xor other.data)
8 bits of the resulting 'Long' value are the same as the bits of this 'UByte' value, whereas the most significant 56 bits are filled with zeros.

*/*n @kotlin.internal.InlineOnly
public inline fun toLong(): Long =
data.toLong() and 0xFF
/** Returns this value. */
*/*n @kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = this
/** Converts this [UByte] value to [UShort]. */
*/*n * The resulting 'UShort' value represents the same numerical value as this 'UByte'.
*/*n * The least significant 8 bits of the resulting 'UShort' value are the same as the bits of this 'UByte' value, whereas the most significant 8 bits are filled with zeros.

*/*n @kotlin.internal.InlineOnly
public inline fun toUShort(): UShort =
UShort(data.toShort() and 0xFF)
/** Converts this [UByte] value to [UInt]. */
*/*n * The resulting 'UInt' value represents the same numerical value as this 'UByte'.
*/*n * The least significant 24 bits are filled with zeros.

*/*n @kotlin.internal.InlineOnly
public inline fun toUInt(): UInt =
UInt(data.toInt() and 0xFF)
/** Converts this [UByte] value to [ULong]. */
*/*n * The resulting 'ULong' value represents the same numerical value as this 'UByte'.
*/*n * The least significant 8 bits of the result 'ULong' value are the same as the bits of this 'UByte' value, whereas the most significant 56 bits are filled with zeros.

*/*n @kotlin.internal.InlineOnly
public inline fun toULong(): ULong =
ULong(data.toLong() and 0xFF)
/** Converts this [UByte] value to [Float]. */
*/*n * The resulting 'Float' value represents the same numerical value as this 'UByte'.
*/*n @kotlin.internal.InlineOnly
public inline fun toFloat(): Float =
this.toInt().toFloat()
/** Converts this [UByte] value to [Double]. */
*/*n * The resulting 'Double' value represents the same numerical value as this 'UByte'.
*/*n @kotlin.internal.InlineOnly
public inline fun toDouble(): Double =
this.toInt().toDouble()
/**
   * The number of bits used to represent an instance of UInt in a binary form.
   */
   public const val SIZE_BITS: Int = 32
   
   /**
       * Compares this value with the specified value for order.
       */
   public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toUInt())
   
   /**
       * Compares this value with the specified value for order.
       */
   public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toUInt())
   
   /**
       * Compares this value with the specified value for order.
       */
   public inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data)
   
   /**
       * Compares this value with the specified value for order.
       */
   public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)
   
   /**
       * Compares this value with the specified value for order.
       */
   @kotlin.internal.InlineOnly
   public inline operator fun compareTo(other: ULong): ULong = this.toULong().compareTo(other)
   
   /**
       * Adds the other value to this value.
       */
   public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt())
   
   /**
       * Adds the other value to this value.
       */
   public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt())
   
   /**
       * Adds the other value to this value.
       */
   public inline operator fun plus(other: UInt): UInt = UInt(this.data.plus(other.data))
   
   /**
       * Adds the other value to this value.
       */
   public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)
   
   /**
       * Subtracts the other value from this value.
       */
   public inline operator fun minus(other: UByte): UInt = this.minus(other.toUInt())
   
   /**
       * Subtracts the other value from this value.
       */
   public inline operator fun minus(other: UShort): UInt = this.minus(other.toUInt())
   
   /**
       * Subtracts the other value from this value.
       */
   public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data))
   
   /**
       * Subtracts the other value from this value.
       */
   public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
   
   /**
       * Multiplies this value by the other value.
       */
   public inline operator fun times(other: UByte): UInt = this.times(other.toUInt())
   
   /**
       * Multiplies this value by the other value.
       */
   public inline operator fun times(other: UShort): UInt = this.times(other.toUInt())
   
   /**
       * Multiplies this value by the other value.
       */
   public inline operator fun times(other: UInt): UInt = UInt(this.data.times(other.data))
   
   /**
       * Multiplies this value by the other value.
       */
   public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
   
   /**
       * Calculates the remainder of dividing this value by the other value.
       */
   public inline operator fun rem(other: UByte): UInt = this.rem(other.toUInt())
   
   /**
       * Calculates the remainder of dividing this value by the other value.
       */
   public inline operator fun rem(other: UShort): UInt = this.rem(other.toUInt())
   
   /**
       * Calculates the remainder of dividing this value by the other value.
       */
   public inline operator fun rem(other: UInt): UInt = uintRemainder(this, other)
   
   /**
       * Calculates the remainder of dividing this value by the other value.
       */
   public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
   
   /**
       * Increments this value.
       */
   public inline operator fun inc(): UInt = UInt(data.inc())
   
   /**
       * Decrements this value.
       */
   public inline operator fun dec(): UInt = UInt(data.dec())

Open Source Used In DNAC Platform Arica  13123
public inline operator fun dec(): UInt = UInt(data.dec())

/** Creates a range from this value to the specified [other] value. */
@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UInt): UIntRange =
UIntRange(this, other)

/** Shifts this value left by the [bitCount] number of bits. */
@kotlin.internal.InlineOnly
public inline infix fun shl(bitCount: Int): UInt = UInt(data.shl(bitCount))

/** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): UInt = UInt(data.ushr(bitCount))

/** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: UInt): UInt = UInt(this.data and other.data)

/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: UInt): UInt = UInt(this.data or other.data)

/** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: UInt): UInt = UInt(this.data xor other.data)

/** Performs a bitwise NOT operation on this value. */
@kotlin.internal.InlineOnly
public inline infix fun not(): UInt = UInt(~this.data)

/** Converts this [UInt] value to [Byte]. */
@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()

/** Converts this [UInt] value to [Short]. */
@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()

/** Converts this [UInt] value to [Int]. */
@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()

/** Converts this [UInt] value to [Long]. */
@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong() and 0xFFFFFFFF

/** Converts this [UInt] value to [ULong]. */
@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = ULong(data.toLong() and 0xFFFF_FFFF)

/** Converts this [UInt] value to [Double]. */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = data.toDouble()

/** Converts this [UInt] value to [Float]. */
@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = data.toFloat()

/** Converts this [UInt] value to [UByte]. */
@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [UInt] value to [UShort]. */
@kotlin.internal.InlineOnly
public inline fun toUShort(): UShort = data.toUShort()

/** Converts this [UInt] value to [UShort]. */
@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = data.toUInt()

/** Converts this [UInt] value to [UByte]. */
@kotlin.internal.InlineOnly
public inline fun toUByte(): UByte = data.toUByte()

/** Converts this [UInt] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Byte): ByteRange = ByteRange(this, other)

/** Converts this [Short] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Short): ShortRange = ShortRange(this, other)

/** Converts this [Int] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Int): IntRange = IntRange(this, other)

/** Converts this [Long] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Long): LongRange = LongRange(this, other)

/** Converts this [ULong] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)

/** Converts this [Double] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Double): DoubleRange = DoubleRange(this, other)

/** Converts this [Float] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: Float): FloatRange = FloatRange(this, other)

/** Converts this [UShort] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: UShort): UShortRange = UShortRange(this, other)

/** Converts this [UByte] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: UByte): UByteRange = UByteRange(this, other)

/** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: Byte): Byte = this.data and other.data

/** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: Byte): Byte = this.data or other.data

/** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: Byte): Byte = this.data xor other.data

/** Performs a bitwise NOT operation on this value. */
@kotlin.internal.InlineOnly
public inline infix fun not(): Byte = ~this.data

/** Converts this [Byte] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: ByteRange): ByteRange = ByteRange(this, other)

/** Converts this [Short] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: ShortRange): ShortRange = ShortRange(this, other)

/** Converts this [Int] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: IntRange): IntRange = IntRange(this, other)

/** Converts this [Long] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: LongRange): LongRange = LongRange(this, other)

/** Converts this [ULong] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: ULongRange): ULongRange = ULongRange(this, other)

/** Converts this [Double] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: DoubleRange): DoubleRange = DoubleRange(this, other)

/** Converts this [Float] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: FloatRange): FloatRange = FloatRange(this, other)

/** Converts this [UShort] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: UShortRange): UShortRange = UShortRange(this, other)

/** Converts this [UByte] value to [ByteRange]. */
@kotlin.internal.InlineOnly
public inline fun rangeTo(other: UByteRange): UByteRange = UByteRange(this, other)

/** Returns the result of the `UInt` value. */
@kotlin.internal.InlineOnly
public inline operator fun dec(): UInt = this.data.dec()
most significant 24 bits are filled with the sign bit of this value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Byte.toUInt(): UInt = UInt(this.toInt())/*\@n * Converts this [Short] value to [UInt].\@n * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Short`.\@n * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value,\@n * whereas the most significant 16 bits are filled with the sign bit of this value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Short.toUInt(): UInt = UInt(this.toInt())/*\@n * Converts this [Int] value to [UInt].\@n * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Int`.\@n * The resulting `UInt` value has the same binary representation as this `Int` value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Int.toUInt(): UInt = UInt(this)
/**
 * Converts this [Short] value to [UInt].
 *
 * If this value is positive, the resulting `UInt` value represents the same numerical value as this `Long` value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Long.toUInt(): UInt = UInt(this)

/**
 * Converts this [Float] value to [UInt].
 *
 * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Long`.\@n * The resulting `UInt` value is represented by the least significant 32 bits of this `Long` value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Float.toUInt(): UInt = FloatToUInt(this)
/**
 * Converts this [Double] value to [UInt].
 *
 * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Double` value.

*\@SinceKotlin\("1.3\")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun
Double.toUInt(): UInt = DoubleToUInt(this)

// Auto-generated file. DO NOT EDIT!

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline class UShort @PublishedApi internal
constructor(\@PublishedApi internal val data: Short) : Comparable<UShort> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UShort can have.
         */
        public const val MIN_VALUE: UShort = UShort(0)

        /**
         * A constant holding the maximum value an instance of UShort can have.
         */
        public const val MAX_VALUE: UShort = UShort(-1)

        /**
         * The number of bytes used to represent an instance of UShort in a binary form.
         */
        public const val SIZE_BYTES: Int = 2

        /**
         * The number of bits used to represent an instance of UShort in a binary form.
         */
        public const val SIZE_BITS: Int = 16

        @kotlin.internal.InlineOnly
        @Suppress("OVERRIDE_BY_INLINE")
        public override inline fun compareTo(other: UByte): Int = toInt().compareTo(other.toInt())

        @kotlin.internal.InlineOnly
        public inline fun compareTo(other: UShort): Int = toInt().compareTo(other.toInt())

        @kotlin.internal.InlineOnly
        public inline fun compareTo(other: UInt): Int = toInt().compareTo(other.toInt())

        @kotlin.internal.InlineOnly
        public inline fun compareTo(other: ULong): Int =
this.toULong().compareTo(other))

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())
    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UShort): UInt =
        this.toUInt().plus(other.toUInt())
    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong =
        this.toULong().plus(other.toUInt())

    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): UInt =
        this.toUInt().minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): UInt =
        this.toUInt().minus(other.toUInt())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong =
        this.toULong().minus(other.toUInt())

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): UInt =
        this.toUInt().times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): UInt =
        this.toUInt().times(other.toUInt())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong =
        this.toULong().times(other.toUInt())

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): UInt =
        this.toUInt().div(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): UInt =
        this.toUInt().div(other.toUInt())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong =
        this.toULong().div(other.toUInt())

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): UInt =
        this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): UInt =
        this.toUInt().rem(other.toUInt())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong =
        this.toULong().rem(other.toUInt())

    /** Increments this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): UShort =
        UShort(data.inc())
    /** Decrements this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): UShort =
        UShort(data.dec())

    /** Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: UShort): UIntRange =
        UIntRange(this.toUInt(), other.toUInt())

    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: UShort): UShort =
        UShort(this.data and other.data)
    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: UShort): UShort =
        UShort(this.data or other.data)
    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: UShort): UShort =
        UShort(this.data xor other.data)
    /** Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline fun inv(): UShort =
        UShort(data.inv())
    /** Converts this [UShort] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline fun toByte(): Byte =
        data.toByte()
value is less than or equals to [Short.MAX_VALUE], the resulting ’Short’ value represents
* the same numerical value as this ’UShort’. Otherwise the result is negative.
* The resulting ’Short’ value has the same binary representation as this ’UShort’ value.
 */
@kotlin.internal.InlineOnly public inline fun toShort(): Short = data//
*/
* Converts this [UShort] value to [Int].
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
* The least significant 16 bits of the resulting ’Int’ value are the same as the bits of this ’UShort’ value,
* whereas the most significant 16 bits are filled with zeros.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
* The resulting ’Long’ value represents the same numerical value as this ’UShort’.
* The least significant 16 bits of the resulting ’Long’ value are the same as the bits of this ’UShort’ value,
* whereas the most significant 48 bits are filled with zeros.
*/
@kotlin.internal.InlineOnly public inline fun toLong(): Long = data.toLong() and 0xFFFF
*/
* Converts this [UShort] value to [UByte].
* If this value is less than or equals to [UByte.MAX_VALUE], the resulting ’UByte’ value represents
* the same numerical value as this ’UShort’.
* The resulting ’UByte’ value is represented by the least significant 8 bits of this ’UShort’ value.
*/
@kotlin.internal.InlineOnly public inline fun toUByte(): UByte = data.toUByte()
*/
* Returns this value.
*/
@kotlin.internal.InlineOnly public inline fun toUShort(): UShort = UShort(data.toShort())
*/
* Converts this [UShort] value to [UByte].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
* The least significant 16 bits of the resulting ’Int’ value are the same as the bits of this ’UShort’ value,
* whereas the most significant 16 bits are filled with zeros.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [UByte].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
* The least significant 16 bits of the resulting ’Int’ value are the same as the bits of this ’UShort’ value,
* whereas the most significant 16 bits are filled with zeros.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
*/
* Converts this [UShort] value to [Long].
*/
* Converts this [UShort] value to [Int].
*/
* The resulting ’Int’ value represents the same numerical value as this ’UShort’.
*/
@kotlin.internal.InlineOnly public inline fun toInt(): Int = data.toInt() and 0xFFFF
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be

Open Source Used in DNAC Platform Arica  13127


fun containsAll(elements: Collection<T>): Boolean = elements.all { contains(it) } 

override fun iterator():
    return ArrayList().asCollection(elements).iterator()

override fun list(): List<T> = elements.asList()
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <E>
buildList(capacity: Int, @BuilderInference builderAction: MutableList<E>.() -> Unit): List<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE)
    }
    checkBuilderCapacity(capacity)
    return ArrayList<E>(capacity).apply(builderAction)
}

/**
 * Returns an [IntRange] of the valid indices for this
 * collection.
 * @sample samples.collections.Collections.Collections.indicesOfCollection
 */
public val Collection<*>.indices: IntRange
    get() = 0..size - 1

/**
 * Returns the index of the last item in the list or -1 if
 * the list is empty.
 * @sample samples.collections.Collections.Collections.lastIndex
 */
@kotlin.internal.InlineOnly
public inline fun <T> List<T>.lastIndex: Int
    get() = this.size - 1

/**
 * Returns `true` if the collection is not empty.
 * @sample samples.collections.Collections.Collections.isNotEmpty
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isEmpty(): Boolean = !this.isNullOrEmpty

/**
 * Checks if all
 * elements in the specified collection are contained in this collection.
 * @sample samples.collections.Collections.Collections.containsAll
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in
// some cases
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T>
Collection<T>.containsAll(elements: Collection<T>): Boolean = this.containsAll(elements)

internal fun <T>
List<T>.optimizeReadOnlyList() = when (size) {
    0 -> emptyList()
    1 -> listOf(this[0])
    else ->
        this
}

/**
 * Searches this list or its range for the provided 
 * [element] using the binary search algorithm.
 * The
 * list is expected to be sorted into ascending order according to the 
 * given [comparator], otherwise the result is undefined.
 * The
 * insertion point is defined as the index at which the element should be
 * inserted, so that the list (or the specified subrange of list) still remains sorted.
 * @sample
 * @sample
 */
@kotlin.internal.InlineOnly
public inline fun <T : Comparable<T>>
List<T?>.binarySearch(element: T?, fromIndex: Int = 0, toIndex: Int = size): Boolean {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = compareValues(midVal, element)

        if (cmp < 0) {
            low = mid + 1
        } else if (cmp > 0) {
            high = mid - 1
        } else {
            // key found
            return true
        }
    }
    // key not found
    return false
}
value is considered to be less than any non-null value.

*return the index of the element, if it is contained in the list within the specified range; otherwise, the inverted insertion point `(insertion point - 1)`.

The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted according to the specified [comparator].

samples.Collections.Collections.Lists.binarySearchWithComparator

*public fun <T>

List<T>.binarySearch(element: T, comparator: Comparator<in T>, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)
    var low = fromIndex
    var high = toIndex - 1
    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparator.compare(midVal, element)
        if (cmp < 0) {
            low = mid + 1
        } else if (cmp > 0) {
            high = mid - 1
        } else {
            return mid // key found
        }
    }
    return -(low + 1) // key not found
}

* Searches this list or its range for an element having the key returned by the specified [selector] function equal to the provided [key] value using the binary search algorithm.

The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements. otherwise the result is undefined.

If the list contains multiple elements with the specified [key], there is no guarantee which one will be found.

`null` value is considered to be less than any non-null value.

*return the index of the element, if it is contained in the list within the specified range; otherwise, the inverted insertion point `(insertion point - 1)`.

The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.

samples.Collections.Collections.Lists.binarySearchByKey

*inline fun <T, K : Comparable<K>>

List<T>.binarySearchBy(key: K?, fromIndex: Int = 0, toIndex: Int = size, selector: (T) -> K?): Int =
    binarySearch(fromIndex, toIndex) { compareValues(selector(it), key) }

// do not introduce this overload --- too rare

@PublishedApi
@SinceKotlin("1.3")
internal fun throwIndexOverflow() { throw

IllegalArgumentOutOfRangeException("fromIndex ($fromIndex) is greater than toIndex ($toIndex).")
    }

fromIndex < 0 -> throw IndexOutOfBoundsException("fromIndex ($fromIndex) is less than zero.
    
fromIndex > size -> throw IndexOutOfBoundsException("toIndex ($toIndex) is greater than size ($size).
    
Uninital expect fun checkIndexOverflow(index: Int):

Uninital expect fun checkCountOverflow(count: Int):
ArithmeticException("Index overflow has happened.") }\n\nthrowCountOverflow() { throw ArithmeticException("Count overflow has happened.") }

@PublishedApi
@SinceKotlin("1.3")
internal fun throwCountOverflow() { throw ArithmeticException("Count overflow has happened.") }

/*================================================================*/
* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")
@file:UseExperimental(kotlin.experimental.ExperimentalTypeInference::class)

package kotlin.collections

import kotlin.contracts.*

private object EmptyMap : Map<Any?, Nothing>, Serializable {\n    private const val serialVersionUID: Long = 8246714829545688274

    override fun equals(other: Any?): Boolean = other is Map<* , *> && other.isEmpty()\n    override fun hashCode(): Int = 0\n    override fun toString(): String = "{}"

    override val size: Int get() = 0\n    override fun isEmpty(): Boolean = true\n    override fun containsValue(value: Nothing): Boolean = false\n    override fun containsKey(key: Any?): Boolean = false\n    override fun get(key: Any?): Nothing? = null\n    override fun values: Collection<Nothing> get() = EmptyList\n    override fun valuesAt(indexs: IntArray): Collection<Nothing> get() = EmptyList\n    override fun keys: Set<Any?> get() = EmptySet\n    override fun entries: Set<Map.Entry<Any?, Nothing>> get() = EmptySet\n
    private fun readResolve(): Any = EmptyMap
}

/**
 * Returns an empty read-only map of specified type.
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> emptyMap(): Map<K, V> = @Suppress("UNCHECKED_CAST") (EmptyMap as Map<K, V>)

/**
 * Returns a new read-only map with the specified contents, given as a list of pairs
 * where the first value is the key and the second is the value.
 * @sample samples.collections.Maps.Instantiation.mapFromPairs
 */
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =
    if (pairs.size > 0) pairs.toMap(LinkedHashMap(mapCapacity(pairs.size))) else
    emptyMap()\n
/**
 * Returns an empty new [MutableMap].
 * @sample samples.collections.Maps.Instantiation.emptyMutableMap
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <K, V> mutableMapOf(): MutableMap<K, V> = LinkedHashMap()

/**
 * Returns a new [MutableMap] with the specified contents, given as a list of pairs
 * where the first component is the key and the second is the value.
 * @sample samples.collections.Maps.Instantiation.mutableMapFromPairs
 * @sample samples.collections.Maps.Instantiation.emptyMutableMap
 */
public fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns an empty new [HashMap].
 * @sample samples.collections.Maps.Instantiation.emptyHashMap
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <K, V> hashMapOf(): HashMap<K, V> = HashMap<K, V>()

/**
 * Returns a new [HashMap] with the specified contents, given as a list of pairs
 * where the first component is the key and the second is the value.
 * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
 * @sample samples.collections.Maps.Instantiation.hashMapFromPairs
 */
public fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> =
    HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns an empty new [LinkedHashMap].
 * @sample samples.collections.Maps.Instantiation.linkedMapFromPairs
 */
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }
maps one or more keys to the specified `value`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.

* Allows to overcome type-safety restriction of `containsValue` that requires to pass a value of type `V`.
Puts all the given [pairs] into this [MutableMap] with the first component in the pair being the key and the second the value.

*public fun <K, V> MutableMap<K, V> putAll(pairs: Array<Pair<K, V>>): Unit { for ((key, value) in pairs) { put(key, value) } }* Puts all the elements of the given collection into this [MutableMap] with the first component in the pair being the key and the second the value.

*public fun <K, V> ImmutableMap<K, V> putAll(pairs: Iterable<Pair<K, V>>): Unit { for ((key, value) in pairs) { put(key, value) } }* Returns a new map with entries having the keys of the map and the values obtained by applying the [transform] to each entry in this [Map].

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapValues

*public fun <K, V, R> Map<K, V> mapValues(transform: (Map.Entry<K, V>) -> R): Map<K, R> { return mapValuesTo(LinkedHashMap<K, R>(mapCapacity(size)), transform) // .optimizeReadOnlyMap() * Returns a new Map with entries having the keys obtained by applying the [transform] function to each entry in this [Map] and the values of this map.

*public fun <K, V> Map<K, V> mapKeys(transform: (Map.Entry<K, V>) -> V): Map<R, V> { return mapKeysTo(LinkedHashMap<R, V>(mapCapacity(size)), transform) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs with keys matching the given [predicate].

*public fun <K, V> Map<K, V> filterKeys(predicate: (K) -> Boolean): Map<K, V> { return filterKeysTo(LinkedHashMap<K, V>(), predicate) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs with values matching the given [predicate].

*public fun <K, V> Map<K, V> filterValues(predicate: (V) -> Boolean): Map<K, V> { return filterValuesTo(LinkedHashMap<K, V>(), predicate) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs not matching the given [predicate].

*public fun <K, V> Map<K, V> filterNot(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> { return filterNotTo(LinkedHashMap<K, V>(), predicate) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs not matching the given [predicate].

*public fun <K, V> Map<K, V> filterNotTo(destination: Map<K, V>, predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> { return filterNotTo(destination, predicate) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs not matching the given [predicate].

*public fun <K, V> Map<K, V> filterTo(destination: Map<K, V>, predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> { return filterTo(destination, predicate) // .optimizeReadOnlyMap() * Returns a new Map containing all key-value pairs matching the given [predicate].

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapKeys

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapValues

*public fun <K, V> Map<K, V> filterOf* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.filterKeys

*public fun <K, V> Map<K, V> filter* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.filterValues

*public fun <K, V> Map<K, V> filterNot* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.filterNot

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapKeys

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapValues

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapKeys

*public fun <K, V> Map<K, V> of* The returned map preserves the entry iteration order of the original map.

@sample samples.collections.Maps.Transformations.mapValues
same key the last one gets added to the map.

Iterable<Pair<K, V>>.toMap(): Map<K, V> {
    if (this is Collection) { 
        return when (size) {  
            0 -> emptyMap()
            1 -> mapOf(if (this is List) this[0] else iterator().next())
        }
    } else -> toMap(LinkedHashMap<K, V>(mapCapacity(size)))
}

return toMap(LinkedHashMap<K, V>(())).optimizeReadOnlyMap()

Populates and returns the [destination] mutable map with key-value pairs from the given collection of pairs.

putAll(this@toMap) |\n
* Returns the new map containing all key-value pairs from the given array of pairs.

if any of two pairs would have the same key the last one gets added to the map.

Replace or adds an entry to this map from another [map].

if (this.isEmpty()) mapOf(pairs) else LinkedHashMap(this).apply { putAll(pairs) }

Optimizes the read-only map.

Appends or replaces
the given [pair] in this mutable map.

```kotlin
MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {
    putAll(pairs)
}
```

---

```
/**
 * Appends or replaces all pairs from the given collection of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Iterable<Pair<K, V>>) {
    putAll(pairs)
}
```

---

```
/**
 * Appends or replaces all pairs from the given sequence of [pairs] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(pairs: Sequence<Pair<K, V>>, ) {
    putAll(pairs)
}
```

---

```
/**
 * Appends or replaces all pairs from the given [map] in this mutable map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.plusAssign(map: Map<K, V>) {
    putAll(map)
}
```

---

```
/**
 * Returns a map containing all entries of the original map except the entry with the given [key].
 * The returned map preserves the entry iteration order of the original map.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(key) }

```

---

```
/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] collection.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }
```

---

```
/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] array.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }
```

---

```
/**
 * Returns a map containing all entries of the original map except those entries
 * the keys of which are contained in the given [keys] sequence.
 */
@SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }
```

---

```
/**
 * Removes the entry with the given [key] from this mutable map.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<in K, in V>.minusAssign(key: K) {
    remove(key)
}
```

---

```
/**
 * Removes all entries the keys of which are contained in the given [keys] collection from this mutable map.
 */
@SinceKotlin("1.1")
public inline operator fun <K, V> MutableMap<in K, in V>.minusAssign(keys: Iterable<K>) {
    this.keys.removeAll(keys)
}
```

---

```
/**
 * Removes all entries the keys of which are contained in the given [keys] array from this mutable map.
 */
@SinceKotlin("1.1")
public inline operator fun <K, V> MutableMap<in K, in V>.minusAssign(keys: Array<out K>) {
    this.keys.removeAll(keys)
}
```

---

```
/**
 * Removes all entries from the keys of which are contained in the given [keys] sequence from this mutable map.
 */
@SinceKotlin("1.1")
public inline operator fun <K, V> MutableMap<in K, in V>.minusAssign(keys: Sequence<K>) {
    this.keys.removeAll(keys)
}
```

---

```
/**
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
```

---

```
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SetsKt")
@file:UseExperimental(kotlin.experimental.ExperimentalTypeInference::class)
package kotlin.collections

import kotlin.contracts.*

internal object EmptySet : Set<Nothing>, Serializable {
    private const val serialVersionUID: Long = 3406603774387020532

    override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty() && other.isEmpty()
    override fun hashCode(): Int = 0
    override fun toString(): String = "[]"
}
```

---

```
override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun contains(element:
Nothing(): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
    override fun iterator(): Iterator<Nothing> = EmptyIterator
    private fun readResolve(): Any = EmptySet

/**
 * Returns an empty read-only set. The returned set is serializable (JVM).
 */
public fun <T> emptySet(): Set<T> = EmptySet

/**
 * Returns a new read-only set with the given elements.
 * Elements of the set are iterated in the order they were specified.
 * The returned set is serializable (JVM).
 */
@sample samples.collections.Collections.Sets.emptyReadOnlySet
*\n@sample samples.Collections.Sets.emptyReadOnlySet
*\n@sample samples.collections.Collections.Sets.emptyReadOnlySet
*\n@Override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
*/

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
import kotlin.text.*

@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val start: Int
    val isNegative: Boolean
    val limit: Long

    val firstChar = this[0]
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after
        return
    }

    var limitBeforeMul = limitForMaxRadix
    var result = 0L
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        result *= radix
        if (result < limit + digit) return null
        result -= digit
    }

    return if (isNegative) result else -result
}

/**
 * Parses the string as a signed [Long] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? = toLongOrNull(radix = 10)

/**
 * Parses the string as a [Long] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toLongOrNull(): Long? = toLongOrNull(radix = 10)

/**
 * Parses the string as a [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(radix: Int): Byte? = toByteOrNull(radix = 10)

/**
 * Parses the string as a [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)

/**
 * Parses the string as a [Short] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toShortOrNull(radix: Int): Short? = toShortOrNull(radix = 10)

/**
 * Parses the string as a [Short] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toShortOrNull(): Short? = toShortOrNull(radix = 10)

/**
 * Parses the string as a signed [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)

/**
 * Parses the string as a signed [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)
IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
    override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())
}

/**
 * Returns the size of this iterable if it is known, or null otherwise.
 */
@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrNull(): Int? = if (this is Collection<*>) this.size else null

/**
 * Returns the size of this iterable if it is known, or the specified [defaultValue] value otherwise.
 */
@PublishedApi
internal fun <T> Iterable<T>.collectionSizeOrDefault(default: Int): Int = if (this is Collection<*>) this.size else default

/**
 * Returns true when it's safe to convert this collection to a set without changing contains method behavior.
 */
@PublishedApi
internal fun <T> Iterable<T>.safeToConvertToSet(): Boolean

/**
 * Converts this collection to a set, when it's worth so and it doesn't change contains method behavior.
 */
@PublishedApi
internal fun <T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =
    when (this)
    {
        is Set -> this
        is Collection
        -> if (this.safeToConvertToSet()) toHashSet() else this
        else -> if (this.safeToConvertToSet()) toHashSet() else this
    }

/**
 * Converts this collection to a set, when it's worth so and it doesn't change contains method behavior.
 */
@PublishedApi
internal fun <T> Iterable<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> =
    when (this)
    {
        is Set -> this
        is Collection
        -> if (this.safeToConvertToSet()) toHashSet() else this
        else -> if (this.safeToConvertToSet()) toHashSet() else this
    }

/**
 * Given an [iterator] function constructs a [Sequence] that returns values through the [Iterator]
 * provided by that function. The values are evaluated lazily, and the sequence is potentially
 * infinite.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence(crossinline iterator: () -> Iterator<T>): Sequence<T> = object : Sequence<T> {
    override fun iterator(): Iterator<T> = iterator()
}

/**
 * Creates a sequence that returns all elements from this iterator. The sequence is constrained to be iterated only once.
 */
public fun <T> Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()

/**
 * Creates a sequence that returns the specified values.
 */
public fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence() else elements.asSequence()

/**
 * Returns an empty sequence.
 */
public fun <T> emptySequence(): Sequence<T> = EmptySequence

private object EmptySequence : Sequence<Nothing>, DropTakeSequence<Nothing> {
    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun drop(n: Int) = EmptySequence
    override fun take(n: Int) = EmptySequence
}

/**
 * Returns a single list of all elements from all collections in the given collection.
 */
public fun <T> Iterable<Iterable<T>>.flatten(): List<T> {
    val result = ArrayList<T>()
    for (element in this)
    {
        result.addAll(element)
    }
    return result
}

/**
 * Returns a pair of lists, where the first list is built from the first values of each pair from this collection,
 * and the second list is built from the second values of each pair from this collection.
 */
public fun <T, R> Iterable<Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {
    val expectedSize = collectionSizeOrDefault(10)
    val listT = ArrayList<T>(expectedSize)
    val listR = ArrayList<R>(expectedSize)
    for (pair in this)
    {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

*/
/*
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

package kotlin.sequences

/**
 * Returns a sequence that iterates through the elements of this sequence.
 */
@PublishedApi
public fun <T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>): Sequence<T> = sequence {
    val iterator =
    if (hasNext())
        doSomething(iterator)
    else
        defaultValue()
private val sequence1: Sequence<T1>,
private val sequence2: Sequence<T2>,
private val transform: (T1, T2) -> V">n : Sequence<V">n
override fun iterator(): Iterator<V">n
val iterator1 = sequence1.iterator()
val iterator2 = sequence2.iterator()
override fun next(): V">n
return transform(iterator1.next(), iterator2.next())


override fun hasNext(): Boolean
return iterator1.hasNext() && iterator2.hasNext()


private fun calcNext() {
if (iterator.hasNext()) {
val item = iterator.next()


sequence.iterator()
var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
var nextItem: T? = null

private fun ensureItemIterator(): Boolean
if (itemIterator?.hasNext() == false)
itemIterator = null

while (itemIterator == null) {
if (!iterator.hasNext()) {
return false
}
else {
val element = iterator.next()
val nextItemIterator = iterator(transformer(element))
if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

override fun hasNext(): Boolean
return ensureItemIterator()


private fun ensureItemIterator(): Boolean
if (itemIterator?.hasNext() == false)
itemIterator = null

while (itemIterator == null) {
if (!iterator.hasNext()) {
return false
}
else {
val element = iterator.next()
val nextItemIterator = iterator(transformer(element))
if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

override fun hasNext(): Boolean
return ensureItemIterator()


private fun ensureItemIterator(): Boolean
if (itemIterator?.hasNext() == false)
itemIterator = null

while (itemIterator == null) {
if (!iterator.hasNext()) {
return false
}
else {
val element = iterator.next()
val nextItemIterator = iterator(transformer(element))
if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

override fun hasNext(): Boolean
return ensureItemIterator()


private fun ensureItemIterator(): Boolean
if (itemIterator?.hasNext() == false)
itemIterator = null

while (itemIterator == null) {
if (!iterator.hasNext()) {
return false
}
else {
val element = iterator.next()
val nextItemIterator = iterator(transformer(element))
if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

if (nextItemIterator.hasNext()) {
itemIterator = nextItemIterator
return true
}
}

override fun hasNext(): Boolean
return ensureItemIterator()
invokes the function to calculate the next value on each iteration until the function returns `null`. The returned sequence is constrained to be iterated only once.

```
@see constrainOnce
```

**public fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {**
    ```
    return GeneratorSequence(nextFunction, { nextFunction() }).constrainOnce()
```  

**Throws an exception**

Generates a sequence defined by the starting value and the function which is invoked to calculate the next value based on the previous one on each iteration. If `null` value is encountered first, the sequence produces values until it encounters first `null` value. The sequence can be iterated multiple times, each time starting with the seed.

```
@see kotlin.sequences.sequence
```

```
@sample samples.collections.Sequences.Building.generateSequence
```

**fun <T : Any> generateSequence(seed: T?, nextFunction: (T) -> T?): Sequence<T> {**
```
    if (seed == null)
        return EmptySequence
    else
        return GeneratorSequence({ seed }, nextFunction)
```

```
@sample samples.collections.Sequences.Building.generateSequenceWithSeed
```

**fun <T : Any> generateSequence(seedFunction: () -> T?, nextFunction: (T) -> T?): Sequence<T> {**
```
    return GeneratorSequence(seedFunction, nextFunction)
```

```
@sample samples.collections.Sequences.Building.generateSequenceWithLazySeed
```

```
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("PreconditionsKt")
```

```
package kotlin
```

```
import kotlin.contracts.contract
```

```
/**
 * Throws an IllegalArgumentException if the value is false.
 * @sample samples.misc.Preconditions.failRequireWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun require(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    require(value) { "Failed requirement." }
}

```
/**
 * Throws an IllegalArgumentException with the result of calling lazyMessage if the
 * value is false. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failRequireWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun require(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalArgumentException(message.toString())
    } else {
        return value
    }
}

```
/**
 * Throws an IllegalStateException if the value is false. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun check(value: Boolean): Unit {
    contract {
        returns() implies value
    }
    check(value) { "Check failed." }
}

```
/**
 * Throws an IllegalStateException with the result of calling lazyMessage if the value is false.
 * Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun check(value: Boolean, lazyMessage: () -> Any): Unit {
    contract {
        returns() implies value
    }
    if (!value) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

```
/**
 * Throws an IllegalStateException if the value is null. Otherwise returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
public inline fun checkNotNull(value: T?,_lazyMessage: () -> Any): T? {
    contract {
        returns() implies value != null
    }
    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}
```
inline fun <T : Any> checkNotNull(value: T?): T {
    contract {
        returns() implies (value != null)
    }

    return checkNotNull(value) { "Required value was null." }
}

/**
 * Throws an [IllegalStateException] with [lazyMessage] if the [value] is null. Otherwise
 * returns the not null value.
 * @sample samples.misc.Preconditions.failCheckWithLazyMessage
 */
@kotlin.internal.InlineOnly
    contract {
        returns() implies (value != null)
    }

    if (value == null) {
        val message = lazyMessage()
        throw IllegalStateException(message.toString())
    } else {
        return value
    }
}

/**
 * Throws an [IllegalStateException] with the given [message] if the [value] is null. Otherwise
 * returns the not null value.
 * @sample samples.misc.Preconditions.failWithError
 */
@kotlin.internal.InlineOnly
public inline fun error(message: Any): Nothing = throw

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*
import primitiveArrayConcat
import withType
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun <T> Array<out T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun ByteArray.elementAt(index: Int): Byte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun ShortArray.elementAt(index: Int): Short

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun IntArray.elementAt(index: Int): Int

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun LongArray.elementAt(index: Int): Long

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun FloatArray.elementAt(index: Int): Float

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun DoubleArray.elementAt(index: Int): Double

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun BooleanArray.elementAt(index: Int): Boolean

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun CharArray.elementAt(index: Int): Char

/**
 * Returns a [List] that wraps the original array.
 * @sample samples.misc.Collections.Lists.asList
 */
public actual fun <T> Array<out T>.asList(): List<T>
original array.
* 

*An@kotlin.internal.InlineOnly

public actual inline fun ByteString.asList(): List<Byte> {

  return this.unsafeCast<Array<Byte>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun String.asList(): List<Char> {

  return this.unsafeCast<Array<Char>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun ByteArray.asList(): List<Byte> {

  return this.unsafeCast<Array<Byte>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun ShortArray.asList(): List<Short> {

  return this.unsafeCast<Array<Short>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun IntArray.asList(): List<Int> {

  return this.unsafeCast<Array<Int>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun LongArray.asList(): List<Long> {

  return this.unsafeCast<Array<Long>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun FloatArray.asList(): List<Float> {

  return this.unsafeCast<Array<Float>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun DoubleArray.asList(): List<Double> {

  return this.unsafeCast<Array<Double>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun BooleanArray.asList(): List<Boolean> {

  return this.unsafeCast<Array<Boolean>>(this) as List

* 

*An@kotlin.internal.InlineOnly

public actual inline fun CharArray.asList(): List<Char> {

  return object : AbstractList<Char>(), RandomAccess {

    override val size: Int get() = this@asList.size
    override fun isEmpty(): Boolean = this@asList.isEmpty()
    override fun contains(element: Char): Boolean = this@asList.contains(element)
    override fun get(index: Int): Char {
      AbstractList.checkElementIndex(index, size)
      return this@asList[index]
    }
    override fun indexOf(element: Char): Int {
      if ((element as Any?) !is Char) return -1
      return this@asList.indexOf(element)
    }
    override fun lastIndexOf(element: Char): Int {
      if ((element as Any?) !is Char) return -1
      return this@asList.lastIndexOf(element)
    }
  }
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun String.contentEquals(other: String): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Byte>.contentEquals(other: List<Byte>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Short>.contentEquals(other: List<Short>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Int>.contentEquals(other: List<Int>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Long>.contentEquals(other: List<Long>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Float>.contentEquals(other: List<Float>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun List<Double>.contentEquals(other: List<Double>): Boolean {
    definedExternally
}
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun ShortArray.contentEquals(other: ShortArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns `true` if the two specified arrays are
*structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun IntArray.contentEquals(other: IntArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns `true` if the two specified arrays are
*structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun LongArray.contentEquals(other: LongArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns `true` if the two specified arrays are
*structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns `true` if the two specified arrays are
*structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns `true` if the two specified arrays are
*structurally* equal to one another, \n* i.e. contain the same number of the same elements in the same order.
* The elements are compared for equality with the \[\text{equals}][\text{Any.equals}] function.
* For floating point numbers it means that `\text{NaN}` is equal to itself and `-0.0` is not equal to `0.0`.

*An@SinceKotlin("1.1")\n@library("\text{arrayEquals}\n")\npublic actual infix fun CharArray.contentEquals(other: CharArray): Boolean \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the contents of this array
as if it is [List].
*An@SinceKotlin("1.1")\n@library("\text{arrayHashCode}\n")\npublic actual fun <T> Array<out T>.contentHashCode(): Int \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the contents of this array.
*An@SinceKotlin("1.1")\n@library("\text{arrayHashCode}\n")\npublic actual fun ShortArray.contentHashCode(): Int \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the contents of this array.
*An@SinceKotlin("1.1")\n@library("\text{arrayHashCode}\n")\npublic actual fun IntArray.contentHashCode(): Int \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the contents of this array.
*An@SinceKotlin("1.1")\n@library("\text{arrayHashCode}\n")\npublic actual fun LongArray.contentHashCode(): Int \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the contents of this array.
*An@SinceKotlin("1.1")\n@library("\text{arrayHashCode}\n")\npublic actual fun DoubleArray.contentHashCode(): Int \n{\ndefinedExternally\n}*\n**\n* Returns a hash code based on the
contents of this array as if it is [List].

```kotlin
public actual fun BooleanArray.contentHashCode(): Int {
    definedExternally
}
```

```kotlin
public actual fun CharArray.contentHashCode(): Int {
    definedExternally
}
```

```kotlin
public actual fun <T> Array<out T>.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun ByteArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun ShortArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun IntArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun LongArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun FloatArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun DoubleArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun BooleanArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun CharArray.contentToString(): String {
    definedExternally
}
```

```kotlin
public actual fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```

```kotlin
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}
```


```kotlin
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0,
startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this.unsafeCast<Array<Byte>>(),
destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
* Copies this array or its subrange into the [destination] array and returns that array.
* 
* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
* 
* @param destination the array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* 
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
* or when that index is out of the [destination] array indices range.
* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* 
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset],
* or when that index is out of the [destination] array indices range.

```
destination.unsafeCast<Array<Long>>(destinationOffset, startIndex, endIndex)in return destination

* Copies this array or its subrange into the [destination] array and returns that array.
* @param destination the array to copy to
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @return the [destination] array.

*\@SinceKotlin("1.3")\@kotlin.internal.InlineOnly
*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n*public actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray
* Copies this array or its subrange into the [destination] array and returns that array.
* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
* @param destination the array to copy to
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @return the [destination] array.

*\@SinceKotlin("1.3")\@kotlin.internal.InlineOnly
*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n*public actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray
* Copies this array or its subrange into the [destination] array and returns that array.
* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
* @param destination the array to copy to
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @return the [destination] array.

*\@SinceKotlin("1.3")\@kotlin.internal.InlineOnly
*\@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\n*public actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray
* Copies this array or its subrange into the [destination] array and returns that array.
* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
* @param destination the array to copy to
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
* @return the [destination] array.
starting at the specified [destinationOffset].\n * or when that index is out of the [destination] array indices range.\n * \n * @return the [destination] array.\n */
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)\n    return destination\n}

/**
* Returns new array which is a copy of the original array.\n* @return the \[destination\] array.\n*/
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun CharArray.copyOf(): CharArray {
    return withType("CharArray", this.asDynamic().slice())\n}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.\n * In the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @return the \[destination\] array.\n */
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun CharArray.copyOf(newSize: Int): CharArray {
    return fillFrom(this, CharArray(newSize))\n}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.\n * In the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @return the \[destination\] array.\n */
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun CharArray.copyOf(newSize: Int): CharArray {
    return resizePrimitiveCopyOf(newSize)\n}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.\n * In the original array, the copy array is truncated to the [newSize].\n * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.\n * \n * @return the \[destination\] array.\n */
@SinceKotlin("1.3")\n@kotlin.internal.InlineOnly\n@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun CharArray.copyOf(newSize: Int): CharArray {
    return resizePrimitiveCopyOf(newSize)\n}
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

*\npublic actual fun IntArray.copyOf(newSize: Int): IntArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return fillFrom(this, IntArray(newSize))\n  *\n} *\n
public actual fun LongArray.copyOf(newSize: Int): LongArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return withType("LongArray", arrayCopyResize(this, newSize, 0L))\n  *\n} *\n
public actual fun FloatArray.copyOf(newSize: Int): FloatArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return fillFrom(this, FloatArray(newSize))\n} *\n
public actual fun DoubleArray.copyOf(newSize: Int): DoubleArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return fillFrom(this, DoubleArray(newSize))\n} *\n
public actual fun BooleanArray.copyOf(newSize: Int): BooleanArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return withType("BooleanArray", arrayCopyResize(this, newSize, false))\n} *\n
public actual fun CharArray.copyOf(newSize: Int): CharArray { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return withType("CharArray", fillFrom(this, CharArray(newSize)))\n} *\n
public actual fun <T> Array<out T>.copyOf(newSize: Int) { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return arrayCopyResize(this, newSize, null)\n} *\n
abstract class AbstractList<out T> { 'n require(newSize >= 0) { "Invalid new array size: $newSize." }\n  return arrayCopyResize(this, newSize, null)\n  *\n  return fillFrom(this, ArrayRange(fromIndex, toIndex))\n  *\n  return withType("ArrayRange", fromIndex, toIndex)\n  *\n} *\n
open fun <T> copyOfRange(fromIndex: Int, toIndex: Int): Array<T> { 'n AbstractList.checkRangeIndexes(fromIndex,
toIndex, size) * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Returns a new array which is a copy of the specified range of the original array. * @param fromIndex the start of the range (inclusive), must be in 0..array.size * @param toIndex the end of the range (exclusive), must be in fromIndex..array.size * *<public actual fun CharArray.copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    return this.asDynamic().slice(fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value. * @param fromIndex the start of the range (inclusive), 0 by default. * @param toIndex the end of the range (exclusive), size of this array by default. * @throws IndexOutOfBoundsException if fromIndex is less than zero or toIndex is greater than the size of this array. * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
*/
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {
    this.asDynamic().fill(element, fromIndex, toIndex)
}
is greater than the size of this array.

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
    * @param fromIndex the start of the range (inclusive), 0 by default.
    * @param toIndex the end of the range (exclusive), size of this array by default.
    * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
    * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
}
```

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
    * @param fromIndex the start of the range (inclusive), 0 by default.
    * @param toIndex the end of the range (exclusive), size of this array by default.
    * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
    * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
}
```

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
    * @param fromIndex the start of the range (inclusive), 0 by default.
    * @param toIndex the end of the range (exclusive), size of this array by default.
    * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
    * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
}
```

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
    * @param fromIndex the start of the range (inclusive), 0 by default.
    * @param toIndex the end of the range (exclusive), size of this array by default.
    * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
    * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
}
```

```kotlin
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex);
    * @param fromIndex the start of the range (inclusive), 0 by default.
    * @param toIndex the end of the range (exclusive), size of this array by default.
    * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
    * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
}
is greater than the size of this array.

* @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual inline operator fun <T> Array<out T>.plus(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(element: Byte): ByteArray {
    return plus(byteArrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ShortArray.plus(element: Short): ShortArray {
    return plus(shortArrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun IntArray.plus(element: Int): IntArray {
    return plus(intArrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun LongArray.plus(element: Long): LongArray {
    return plus(longArrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun FloatArray.plus(element: Float): FloatArray {
    return plus(floatArrayOf(element))
}

@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun DoubleArray.plus(element: Double): DoubleArray {
    return plus(doubleArrayOf(element))
}

@SinceKotlin("1.3")
@Suppress("ACTUAL_WITHOUT_EXPECT")
public actual inline operator fun Array<out T>.plus(elements: Collection<T>): Array<T> {
    return arrayPlusCollection(this, elements)
}

@public actual inline operator fun ByteArray.plus(elements: Collection<Byte>): ByteArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

@public actual inline operator fun ShortArray.plus(elements: Collection<Short>): ShortArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

@public actual inline operator fun IntArray.plus(elements: Collection<Int>): IntArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
elements.size), this.size, elements)

public actual operator fun LongArray.plus(elements: Collection<Long>): LongArray {
    return arrayPlusCollection(this, elements)
}

public actual operator fun FloatArray.plus(elements: Collection<Float>): FloatArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

public actual operator fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

public actual operator fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {
    return arrayPlusCollection(this, elements)
}

public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyWith(size + elements.size), this.size, elements)
}

public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return this.asDynamic().concat(elements)
}

public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

public actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

public actual fun IntArray.sort(): Unit {
    definedExternally
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun Array<T>.sort(): Unit {
    definedExternally
}
samples.collections.Arrays.Sorting.sortArray() */
public actual fun LongArray.sort(): Unit {
    if (size > 1) sort {
        a: Long, b: Long -> a.compareTo(b) }
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun LongArray.sort(): Unit {
    if (size > 1) sort {
        a: Long, b: Long -> a.compareTo(b) }
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun ByteArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun ShortArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun FloatArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place according to the natural order of its elements.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other
 * after sorting.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun <T : Comparable<T>> Array<out T>.sort(): Unit {
    if (size > 1) sortArrayWith(this, comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun <T> Array<out T>.sort(comparison: (a: T, b: T) -> Int): Unit {
    if (size > 1) sortArrayWith(this, comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.sort(noinline comparison: (a: Byte, b: Byte) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.sort(noinline comparison: (a: Short, b: Short) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.sort(noinline comparison: (a: Int, b: Int) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.sort(noinline comparison: (a: Long, b: Long) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.sort(noinline comparison: (a: Float, b: Float) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.sort(noinline comparison: (a: Double, b: Double) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.sort(noinline comparison: (a: Char, b: Char) -> Int): Unit {
    asDynamic().sort(comparison)
}

/**
 * Sorts the array in-place according to the order specified by the given [comparison] function.
 * 
 * @sample
 * samples.collections.Arrays.Sorting.sortArray
 */
public actual fun <T> Array<out T>.sortWith(comparator: Comparator<in T>): Unit {
    if (size > 1) sortArrayWith(this, comparator)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun ByteArray.toTypedArray(): Array<Byte> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun ShortArray.toTypedArray(): Array<Short> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun IntArray.toTypedArray(): Array<Int> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun LongArray.toTypedArray(): Array<Long> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun FloatArray.toTypedArray(): Array<Float> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun DoubleArray.toTypedArray(): Array<Double> {
    return js("[
]").slice.call(this)
}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public actual fun CharArray.toTypedArray(): Array<Char> {
    return js("[
]").slice.call(this)
}
object array containing all of the elements of this primitive array.

public actual fun DoubleArray.toTypedArray(): Array<Double> {
    return js("\[
        \]").slice.call(this)
}

public actual fun BooleanArray.toTypedArray(): Array<Boolean> {
    return js("\[
        \]").slice.call(this)
}

public actual fun CharArray.toTypedArray(): Array<Char> {
    return Array(size) { index -> this[index] }
}

/**
 * Compares two values using the specified functions [selectors] to calculate the result
 * of the comparison.
 * The functions are called sequentially, receiving the given values [a] and [b] and return
 * [Comparable] objects. As soon as the [Comparable] instances returned by a function for [a] and [b] values do
 * not compare as equal, the result of that comparison is returned.
 */
public fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>?): Int {
    require(selectors.size > 0)
    return compareValuesByImpl(a, b, selectors)
}

private fun <T> compareValuesByImpl(a: T, b: T, selectors: Array<out (T) -> Comparable<*>?>): Int {
    for (fn in selectors) {
        val v1 = fn(a)
        val v2 = fn(b)
        val diff = compareValues(v1, v2)
        if (diff != 0) return diff
    }
    return 0
}

/**
 * Compares two values using the specified [selector] function to calculate the result of the comparison.
 * The function is applied to the given values [a] and [b] and return [Comparable] objects.
 * The result of comparison of these [Comparable] instances is returned.
 */
public inline fun <T> compareValuesBy(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValues(selector(a), selector(b))
}

/**
 * Compares two values using the specified [selector] function to transform values which are then being compared
 * with the given [comparator].
 */
public inline fun <T, K> compareValuesBy(a: T, b: T, comparator: Comparator<in K>, selector: (T) -> K): Int {
    return comparator.compare(selector(a), selector(b))
}

public fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {
    if (a === b) return 0
    if (a == null) return -1
    if (b == null) return 1
    @Suppress("UNCHECKED_CAST")
    return (a as Comparable<Any>).compareTo(b)
}

public fun <T> compareBy(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    require(selectors.size > 0)
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}

public inline fun <T> compareBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(a, b, selector)
    }

public fun <T> compareByWithSelectors(T: T, vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    require(selectors.size > 0)
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}

@kotlin.internal.InlineOnly
public inline fun <T> compareValuesInline(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValuesBy(a, b, selector)
}

@kotlin.internal.InlineOnly
public inline fun <T> compareValuesWith(a: T, b: T, comparator: Comparator<T>): Int {
    return comparator.compare(a, b)
}

@kotlin.internal.InlineOnly
public inline fun <T> compareValuesByWithComparator(a: T, b: T, comparator: Comparator<T>): Int {
    return comparator.compare(a, b)
}

@Suppress("NOTHING_TO_INLINE")
public inline fun <T> compareValuesByWithSingleSelector(a: T, b: T, selector: (T) -> Comparable<*>?): Int {
    return compareValuesBy(a, b, selector)
}

Open Source Used In DNAC Platform Arica 13159
fun <T, K> compareBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
Comparator { a, b -> compareValuesBy(a, b, comparator, selector) }

/*

Results

* @sample samples.comparisons.Comparisons.compareByDescending

*/
@kotlin.internal.InlineOnly
public inline fun <T> compareByDescending(): Comparator<T> =
Comparator { a, b -> compareValuesBy(b, a, comparator, selector) }

/*

Returns a comparator that compares values by the given 
comparator.

Note that an order of [comparator] is reversed by this wrapper.

*/
@kotlin.internal.InlineOnly
public inline fun <T> compareByDescending(comparator: Comparator<T>): Comparator<T> =
    Comparator { a, b -> compareValuesBy(b, a, comparator) }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

* Note that an order of [comparator] is reversed by this wrapper.

*/
@kotlin.internal.InlineOnly
public inline fun <T> thenBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenBy.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(a, b, comparator, selector)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public inline fun <T> thenBy(comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenBy.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(a, b, comparator, selector)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public inline fun <T> thenByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenByDescending.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(b, a, comparator, selector)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public inline fun <T> thenByDescending(comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenByDescending.compare(a, b)
        if (previousCompare != 0) previousCompare else compareValuesBy(b, a, comparator, selector)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public inline fun <T> thenComparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenComparator.compare(a, b)
        if (previousCompare != 0) previousCompare else comparison(a, b)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public infix fun <T> then(crossinline comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@then.compare(a, b)
        if (previousCompare != 0) previousCompare else comparator.compare(a, b)
    }

/*

* creates a comparator comparing values after the primary comparator defined them equal.

* Uses the function to transform value to a [Comparable] instance for comparison.

*/
@kotlin.internal.InlineOnly
public infix fun <T> thenDescending(crossinline comparator: Comparator<in T>): Comparator<T> =
    Comparator { a, b ->
        val previousCompare = this@thenDescending.compare(a, b)
        if (previousCompare != 0) previousCompare else comparator.compare(a, b)
    }
nullsFirst(comparator: Comparator<T>): Comparator<T> = \n Comparator { a, b ->
 when { \n a === b -> 0
 a == null -> -1
 b == null -> 1
 else -> comparator.compare(a, b) }\n }\n
nullsLast(comparator: Comparator<T>): Comparator<T> = \n Comparator { a, b ->
 when { \n a === b -> 0
 a == null -> 1
 b == null -> -1
 else -> comparator.compare(a, b) }\n }

ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {
 override fun compare(a: T, b: T): Int = comparator.compare(b, a)
 }

NothingImplemented(message: String = "An operation is not implemented.")

package kotlin

import kotlin.contracts.*

/**
 * An exception is thrown to indicate that a method body remains to be implemented.
 */
public class NotImplementedError(message: String = "An operation is not implemented.") :

    @file:kotlin.jvm.JvmMultifileClass
    @file:kotlin.jvm.JvmName("StandardKt")
    package kotlin

    import kotlin.contracts.*

    /**
     * An exception is thrown to indicate that a method body remains to be implemented.
     */
    public class NotImplementedError(message: String = "An operation is not implemented.") :

        @file:kotlin.jvm.JvmMultifileClass
        @file:kotlin.jvm.JvmName("StandardKt")
        package kotlin

        import kotlin.contracts.*

        /**
         * An exception is thrown to indicate that a method body remains to be implemented.
         */
        public class NotImplementedError(message: String = "An operation is not implemented.") :

            @file:kotlin.jvm.JvmMultifileClass
            @file:kotlin.jvm.JvmName("StandardKt")
            package kotlin

            import kotlin.contracts.*

            /**
             * An exception is thrown to indicate that a method body remains to be implemented.
             */
            public class NotImplementedError(message: String = "An operation is not implemented.") :

                @file:kotlin.jvm.JvmMultifileClass
                @file:kotlin.jvm.JvmName("StandardKt")
                package kotlin

                import kotlin.contracts.*

                /**
                 * An exception is thrown to indicate that a method body remains to be implemented.
                 */
                public class NotImplementedError(message: String = "An operation is not implemented.") :

                    @file:kotlin.jvm.JvmMultifileClass
                    @file:kotlin.jvm.JvmName("StandardKt")
                    package kotlin

                    import kotlin.contracts.*

                    /**
                     * An exception is thrown to indicate that a method body remains to be implemented.
                     */
                    public class NotImplementedError(message: String = "An operation is not implemented.") :

                        @file:kotlin.jvm.JvmMultifileClass
                        @file:kotlin.jvm.JvmName("StandardKt")
                        package kotlin

                        import kotlin.contracts.*

                        /**
                         * An exception is thrown to indicate that a method body remains to be implemented.
                         */
                        public class NotImplementedError(message: String = "An operation is not implemented.") :

                            @file:kotlin.jvm.JvmMultifileClass
                            @file:kotlin.jvm.JvmName("StandardKt")
                            package kotlin

                            import kotlin.contracts.*

                            /**
                             * An exception is thrown to indicate that a method body remains to be implemented.
                             */
                            public class NotImplementedError(message: String = "An operation is not implemented.") :

                                @file:kotlin.jvm.JvmMultifileClass
                                @file:kotlin.jvm.JvmName("StandardKt")
                                package kotlin

                                import kotlin.contracts.*

                                /**
                                 * An exception is thrown to indicate that a method body remains to be implemented.
                                 */
                                public class NotImplementedError(message: String = "An operation is not implemented.") :

                                    @file:kotlin.jvm.JvmMultifileClass
                                    @file:kotlin.jvm.JvmName("StandardKt")
                                    package kotlin

                                    import kotlin.contracts.*

                                    /**
                                     * An exception is thrown to indicate that a method body remains to be implemented.
                                     */
                                     public class NotImplementedError(message: String = "An operation is not implemented.") :
Error(message)\n\n**\n* Always throws [NotImplementedError] stating that operation is not implemented.\n*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun TODO(): Nothing = throw NotImplementedException()\n*\n*\n* Always throws [NotImplementedError] stating that operation is not implemented.\n*\n* \n* \n* \n* @param reason a string explaining why the implementation is missing.\n*\n*\n*/

@kotlin.internal.InlineOnly\n*\npublic inline fun TODO(): Nothing = throw NotImplementedError()

/**
 * Always throws [NotImplementedError] stating that operation is not implemented.
 *
 * @param reason a string
explaining why the implementation is missing.
 */
@kotlin.internal.InlineOnly\n*\npublic inline fun TODO(reason: String): Nothing = throw NotImplementedError("An operation is not implemented: $reason")

/**
 * Calls
the specified function [block] and returns its result.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#run).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <R> run(block: () -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    return block()\n}

/**
 * Calls
the specified function [block] with `this` value as its receiver and returns its result.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#apply).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <T> T.apply(block: T.() -> Unit): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    block()\n    return this
}

/**
 * Calls
the specified function [block] with `this` value as its argument and returns `this` value.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#also).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <T> T.also(block: (T) -> Unit): T {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    block(this)\n    return this
}

/**
 * Calls
the specified function [block] with `this` value as its argument and returns its result.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#let).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <T, R> T.let(block: (T) -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)\n    }\n    return block(this)
}

/**
 * Returns `this` value if it satisfies the given [predicate] or `null`, if it doesn’t.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#takeIf).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <T> T.takeIf(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)\n    }\n    return if (predicate(this)) this else null
}

/**
 * Returns `this` value if it _does not_ satisfy the given [predicate] or `null`, if it does.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#takeUnless).
*\n*\n@kotlin.internal.InlineOnly\n*\npublic inline fun <T> T.takeUnless(predicate: (T) -> Boolean): T? {
    contract {
        callsInPlace(predicate, InvocationKind.EXACTLY_ONCE)\n    }\n    return if (!predicate(this)) this else null
}

/**
 * Executes the given function [action] specified number of [times].\n*\n* A zero-based index of current iteration is passed as a parameter to [action].\n*\n* @sample
samples.miss.ControlFlow.repeat\n*/
@kotlin.internal.InlineOnly\n*\npublic inline fun repeat(times: Int, action: (Int) -> Unit) {
    contract {
        callsInPlace(action)\n    }\n    for (index in 0 until times) {
        action(index)
    }
}

/**
 * Returns the greater of two values.\n*\n* For detailed usage information see the documentation for [scope functions](https://kotlinlang.org/docs/reference/scope-functions.html#max).
*\n*\n@SinceKotlin("1.1")\npublic actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}

*/

//
// NOTE: THIS FILE IS AUTO-GENERATED by the
GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.js.*

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}
/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Byte, b: Byte): Byte { return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>() }\n* Returns the greater of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Short, b: Short): Short { return Math.max(a.toInt(), b.toInt()).unsafeCast<Short>() }\n* Returns the greater of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Int, b: Int): Int { return Math.max(a, b) }\n* Returns the greater of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Long, b: Long): Long { return if (a >= b) a else b }\n* Returns the greater of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Float, b: Float): Float { return Math.max(a, b) }\n* Returns the greater of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun maxOf(a: Double, b: Double): Double { return Math.max(a, b) }\n* Returns the greater of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Byte, b: Byte): Byte { return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>() }\n* Returns the smaller of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Short, b: Short): Short { return Math.min(a.toInt(), b.toInt()).unsafeCast<Short>() }\n* Returns the smaller of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Int, b: Int): Int { return Math.min(a, b) }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Long, b: Long): Long { return if (a <= b) a else b }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Float, b: Float): Float { return Math.min(a, b) }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Double, b: Double): Double { return Math.min(a, b) }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Byte, b: Byte): Byte { return Math.min(a.toInt(), b.toInt()).unsafeCast<Byte>() }\n* Returns the smaller of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Short, b: Short): Short { return Math.min(a.toInt(), b.toInt()).unsafeCast<Short>() }\n* Returns the smaller of two values.*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Int, b: Int): Int { return Math.min(a, b) }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual inline fun minOf(a: Long, b: Long): Long { return if (a <= b) a else b }\n* Returns the smaller of two values.\n*/

/*\n@SinceKotlin("1.1")\n@kotlin.internal.InlineOnly
@ Suppress("DEPRECATION_ERROR")\npublic actual
inline fun minOf(a: Float, b: Float): Float {\n return Math.min(a, b)\n}\n\n/**\n * Returns the smaller of two
values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Double, b: Double): Double {\n return Math.min(a, b)\n}\n\n/**\n * Returns the smaller of
three values.\n */\n@SinceKotlin(\"1.1\")\npublic actual fun <T : Comparable<T>> minOf(a: T, b: T, c: T): T {\n
return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Byte, b: Byte, c: Byte): Byte {\n return Math.min(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Byte>()\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Short, b: Short, c: Short): Short {\n return Math.min(a.toInt(), b.toInt(),
c.toInt()).unsafeCast<Short>()\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Int, b: Int, c: Int): Int {\n return Math.min(a, b, c)\n}\n\n/**\n * Returns the smaller of three
values.\n */\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\npublic actual inline fun minOf(a: Long, b: Long,
c: Long): Long {\n return minOf(a, minOf(b, c))\n}\n\n/**\n * Returns the smaller of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Float, b: Float, c: Float): Float {\n return Math.min(a, b, c)\n}\n\n/**\n * Returns the smaller
of three values.\n
*/\n@SinceKotlin(\"1.1\")\n@kotlin.internal.InlineOnly\n@Suppress(\"DEPRECATION_ERROR\")\npublic actual
inline fun minOf(a: Double, b: Double, c: Double): Double {\n return Math.min(a, b, c)\n}\n\n","/*\n * Copyright
2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed
by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"CollectionsKt\")\n\npackage
kotlin.collections\n\n//\n// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt\n// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib\n//\n\nimport kotlin.random.*\nimport
kotlin.ranges.contains\nimport kotlin.ranges.reversed\n\n/**\n * Returns 1st *element* from the list.\n * \n *
Throws an [IndexOutOfBoundsException] if the size of this list is less than 1.\n
*/\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component1(): T {\n return
get(0)\n}\n\n/**\n * Returns 2nd *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the
size of this list is less than 2.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T>
List<T>.component2(): T {\n return get(1)\n}\n\n/**\n * Returns 3rd *element* from the list.\n * \n * Throws an
[IndexOutOfBoundsException] if the size of this list is less than 3.\n */\n@kotlin.internal.InlineOnly\npublic inline
operator fun <T> List<T>.component3(): T {\n return get(2)\n}\n\n/**\n * Returns 4th *element* from the list.\n
* \n * Throws an [IndexOutOfBoundsException] if the size of this list is less than 4.\n
*/\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> List<T>.component4(): T {\n return
get(3)\n}\n\n/**\n * Returns 5th *element* from the list.\n * \n * Throws an [IndexOutOfBoundsException] if the
size of this list is less than 5.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T>
List<T>.component5(): T {\n return get(4)\n}\n\n/**\n * Returns `true` if [element] is found in the collection.\n
*/\npublic operator fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.contains(element: T): Boolean {\n if
(this is Collection)\n
return contains(element)\n return indexOf(element) >= 0\n}\n\n/**\n * Returns an
element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this
collection.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n */\npublic fun <T>
Iterable<T>.elementAt(index: Int): T {\n if (this is List)\n
return get(index)\n return
elementAtOrElse(index) { throw IndexOutOfBoundsException(\"Collection doesn't contain element at index
$index.\") }\n}\n\n/**\n * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if
the [index] is out of bounds of this list.\n * \n * @sample samples.collections.Collections.Elements.elementAt\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <T> List<T>.elementAt(index: Int): T {\n return

Open Source Used In DNAC Platform Arica 13164


get(index) * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this collection.

```kotlin
samples.collections.Collections.Elements.elementAtOrElse

Iterable<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    if (this is List)
        return this.getOrElse(index, defaultValue)
    if (index < 0)
        return defaultValue(index)
    val iterator = iterator() val count = 0
    while (iterator.hasNext())
        if (index == count++)
            return iterator.next()
    return defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this list.
 */
public fun <T>
    List<T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex)
        get(index)
    else
defaultValue(index)
}
```

```kotlin
/**
 * Returns an element at the given [index] or null if the [index] is out of bounds of this list.
 */
public fun <T>
    List<T>.getOrNull(index: Int): T? {
    return if (index >= 0 && index <= lastIndex)
        get(index)
    else null
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or null if no such element was found.
 */
public inline fun <T> Iterable<T>.find(predicate: (T) -> Boolean): T? {
    return firstOrNull(predicate)
}
```

```kotlin
/**
 * Returns the last element matching the given [predicate], or null if no such element was found.
 */
public fun <T>
    List<T>.findLast(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
}
```

```kotlin
/**
 * Returns first element.
 * @throws NoSuchElementException if the collection is empty.
 */
public fun <T>
    Iterable<T>.first(): T {
    when (this) {
        is List -> return this.first()
        else -> {
            val iterator = iterator() val count = 0
            while (iterator.hasNext())
                if (index == count++)
                    return iterator.next()
            throw NoSuchElementException("Collection is empty.")
        }
    }
}
```

```kotlin
/**
 * Returns first element.
 * @throws NoSuchElementException if the list is empty.
 */
public fun <T> List<T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("List is empty.")
    return this[0]
}
```

```kotlin
/**
 * Returns the first element matching the given [predicate], or null if element was not found.
 */
public inline fun <T> Iterable<T>.firstOrNull(predicate: (T) -> Boolean): T? {
    for (element in this) if (predicate(element)) return element
    return null
}
```
Open Source Used In DNAC Platform Arica 13166

```kotlin
/**
 * Returns the last element matching the given [predicate], or -1 if the collection does not contain such element.
 * @throws [NoSuchElementException] if the collection is empty.
 */
public inline fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns index of the first element matching the given [predicate], or -1 if the collection does not contain such element.
 * @throws [NoSuchElementException] if the collection is empty.
 */
public inline fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.indexOf(element: T): Int {
    if (this is List) return this.indexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

/**
 * Returns the last index of [element], or -1 if the list does not contain such element.
 * @throws [NoSuchElementException] if the list is empty.
 */
public fun <@kotlin.internal.OnlyInputTypes T> List<T>.lastIndexOf(element: T): Int {
    if (this is List) return this.lastIndexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

```

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> Collection<T>.randomOrNull(random: Random): T? {
    if (isEmpty())
        return null
    return elementAt(random.nextInt(size))
}

/**
 * Returns the single element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> Iterable<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    var last: T? = null
    for (element in this) {
        if (predicate(element)) {
            last = element
        }
    }
    return last
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T> List<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    val iterator = this.iterator()
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (predicate(element))
            return element
    }
    return null
}

/**
 * Returns the single element matching the given [predicate], or throws an exception if there is no or more than one matching element.
 */
public inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return toList()
    var list: ArrayList<T>
    if (this is Collection<*>) {
        val resultSize = size - n
        if (resultSize <= 0)
            return emptyList()
        if (resultSize == 1)
            return listOf(last())
        list = ArrayList<T>(resultSize)
    }
    @Suppress("UNCHECKED_CAST")
    return list
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> Collection<T>.random(random: Random): T {
    if (isEmpty())
        throw NoSuchElementException("Collection is empty.")
    return elementAt(random.nextInt(size))
}

/**
 * Returns a random element from this collection, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.randomOrNull(): T? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this collection using the specified source of randomness, or `null` if this collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> Collection<T>.randomOrNull(random: Random): T? {
    if (isEmpty())
        return null
    return elementAt(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the collection is empty or has more than one element.
 */
@SinceKotlin("1.3")
public fun <T> Iterable<T>.single(): T {
    for (element in this) {
        if (predicate(element))
            return element
    }
    if (!found) throw NoSuchElementException("Collection contains no element matching the given predicate.")
    return element
}

/**
 * Returns the single element, or throws an exception if the collection is empty or has more than one element.
 */
@SinceKotlin("1.3")
public fun <T> List<T>.single(): T {
    return when (size) {
        0 -> throw NoSuchElementException("List is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("List has more than one element.")
    }
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public fun <T> List<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("List contains no element matching the given predicate.")
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 */
public fun <T> Iterable<T>.drop(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return toList()
    val list: ArrayList<T>
    if (this is Collection<*>) {
        val resultSize = size - n
        if (resultSize <= 0)
            return emptyList()
        if (resultSize == 1)
            return listOf(last())
        list = ArrayList<T>(resultSize)
    }
    @Suppress("UNCHECKED_CAST")
    return list
}
if (this is List<T>) {
    if (this is RandomAccess) {
        for (index in n until size)
            list.add(this[index])
    } else {
        for (item in listIterator(n))
            list.add(item)
    }
    return list
}
else {
    list = ArrayList<T>()
}

var count = 0
for (item in this) {
    if (count >= n) list.add(item) else ++count
}
return list.optimizeReadOnlyList()

/**
 * Returns a list containing all elements except last [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample samples.collections.Collections.Transformations.dropLast
 */
public fun <T> List<T>.dropLast(n: Int): List<T> {
    if (!isEmpty()) {
        val iterator = listIterator(size)
        while (iterator.hasPrevious()) {
            if (!predicate(iterator.previous())) {
                return take(iterator.nextIndex() + 1)
            }
        }
    }
    return emptyList()
}

/**
 * Returns a list containing all elements except last elements that satisfy the given
 * [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the
 * element itself and returns the result of predicate evaluation on the element.
 */
public inline fun <T> List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {
    val iterator = listIterator(size)
    while (iterator.hasNext()) {
        if (!predicate(iterator.next())) {
            return take(iterator.nextIndex() + 1)
        }
    }
    return emptyList()
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the
 * element itself and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Iterable<T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @param [predicate] function that takes the index of an element and the
 * element itself and returns the result of predicate evaluation on the element.
 */
public inline fun <T> Iterable<T>.dropIndexedWhile(predicate: (index: Int, T) -> Boolean): List<T> {
    for (index, element) in this
        if (predicate(index, element))
            list.add(element)
    return list
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * 
 * @param [predicate] function that takes the index of an element and the
 * element itself and returns the result of predicate evaluation on the element.
 */
public inline fun <T, C : MutableCollection<in T>> Iterable<T>.dropIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed {
        if (predicate(index, it))
            destination.add(it)
    }
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> {
    return list.slice(indices)
}
sort order of the value returned by specified [selector] function.
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

Returns a list of all elements sorted descending according to their natural sort order.
The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

- Returns a list of all elements sorted according to the specified comparator.

- The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
samples.collections.Collections.Transformations.associateByWithValueTransform

*/

public inline fun <T, K, V> Iterable<T>.associateByWith(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> { val capacity = 
    Map<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)) return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform) } *\n
public inline fun <T, K, V>
    Iterable<T>.associateByWith(destination: M, keySelector: (T) -> K): M { for (element in this) { destination.put(keySelector(element), element) } return destination } *\n
public inline fun <T, K, V, M : MutableMap<K, V>> Iterable<T>.associateTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

public inline fun <K, V> Iterable<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16))
    return associateWithTo(result, valueSelector)
}

public inline fun <K, V, M : MutableMap<K, V>> Iterable<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

public fun <T, C : MutableCollection<T>> Iterable<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

public fun <T> Iterable<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(collectionSizeOrDefault(12))))
}

public fun <T> Iterable<T>.toList(): List<T> {
    if (this is Collection) {
        return when (size) {
            0 -> emptyList()
            1 -> listOf(if (this is List) get(0) else iterator().next())
            else -> this.toMutableList()
        }
    }
    return this.toMutableList().optimizeReadOnlyList()
}

public fun <T> Collection<T>.toMutableList(): MutableList<T> {
    return ArrayList(this)
}

public fun <T> Collection<T>.toSet(): Set<T> {
    return this.toMutableList().toSet()
}

public fun <T> Iterable<T>.iterator(): Iterator<T> {
    return this.iterator()
}

public fun <T> Iterator<T>.next(): T {
    return this.next()
}

public fun <T> Iterator<T>.hasNext(): Boolean {
    return this.hasNext()
}

public fun <T> Iterator<T>.remove(): Unit {
    return this.remove()
}

public fun <T> Iterator<T>.filter(): Filter<T> {
    return this.filter()
}

public fun <T> Iterator<T>.filterNot(): FilterNot<T> {
    return this.filterNot()
}

public fun <T> Iterator<T>.filterIsInstance(): FilterIsInstance<T> {
    return this.filterIsInstance()
}

public fun <T> Iterator<T>.filterNotIsInstance(): FilterNotIsInstance<T> {
    return this.filterNotIsInstance()
}

public fun <T> Iterator<T>.filterNull(): FilterNull<T> {
    return this.filterNull()
}

public fun <T> Iterator<T>.filterIsNotNull(): FilterIsNotNull<T> {
    return this.filterIsNotNull()
}

public fun <T> Iterator<T>.filterIsDefault(): FilterIsDefault<T> {
    return this.filterIsDefault()
}

public fun <T> Iterator<T>.filterIsNotDefault(): FilterIsNotDefault<T> {
    return this.filterIsNotDefault()
}

public fun <T> Iterator<T>.filterIsNot(): FilterIsNot<T> {
    return this.filterIsNot()
}

public fun <T> Iterator<T>.filterIs(): FilterIs<T> {
    return this.filterIs()
}

public fun <T> Iterator<T>.filterIsNullOrEmpty(): FilterIsNullOrEmpty<T> {
    return this.filterIsNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsNotEmpty(): FilterIsNotEmpty<T> {
    return this.filterIsNotEmpty()
}

public fun <T> Iterator<T>.filterIsNotNullOrEmpty(): FilterIsNotNullOrEmpty<T> {
    return this.filterIsNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsNotNullAndNotEmpty(): FilterIsNotNullAndNotEmpty<T> {
    return this.filterIsNotNullAndNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNull(): FilterIsAnyNull<T> {
    return this.filterIsAnyNull()
}

public fun <T> Iterator<T>.filterIsAnyNotNull(): FilterIsAnyNotNull<T> {
    return this.filterIsAnyNotNull()
}

public fun <T> Iterator<T>.filterIsAnyDefault(): FilterIsAnyDefault<T> {
    return this.filterIsAnyDefault()
}

public fun <T> Iterator<T>.filterIsAnyNotDefault(): FilterIsAnyNotDefault<T> {
    return this.filterIsAnyNotDefault()
}

public fun <T> Iterator<T>.filterIsAnyNot(): FilterIsAnyNot<T> {
    return this.filterIsAnyNot()
}

public fun <T> Iterator<T>.filterIsAny(): FilterIsAny<T> {
    return this.filterIsAny()
}

public fun <T> Iterator<T>.filterIsAnyNullOrEmpty(): FilterIsAnyNullOrEmpty<T> {
    return this.filterIsAnyNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotEmpty(): FilterIsAnyNotEmpty<T> {
    return this.filterIsAnyNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullOrEmpty(): FilterIsAnyNotNullOrEmpty<T> {
    return this.filterIsAnyNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullAndNotEmpty(): FilterIsAnyNotNullAndNotEmpty<T> {
    return this.filterIsAnyNotNullAndNotEmpty()
}

public fun <T> Iterator<T>.filterIsAny(): FilterIsAny<T> {
    return this.filterIsAny()
}

public fun <T> Iterator<T>.filterIsAnyNullOrEmpty(): FilterIsAnyNullOrEmpty<T> {
    return this.filterIsAnyNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotEmpty(): FilterIsAnyNotEmpty<T> {
    return this.filterIsAnyNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullOrEmpty(): FilterIsAnyNotNullOrEmpty<T> {
    return this.filterIsAnyNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullAndNotEmpty(): FilterIsAnyNotNullAndNotEmpty<T> {
    return this.filterIsAnyNotNullAndNotEmpty()
}

public fun <T> Iterator<T>.filterIsAny(): FilterIsAny<T> {
    return this.filterIsAny()
}

public fun <T> Iterator<T>.filterIsAnyNullOrEmpty(): FilterIsAnyNullOrEmpty<T> {
    return this.filterIsAnyNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotEmpty(): FilterIsAnyNotEmpty<T> {
    return this.filterIsAnyNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullOrEmpty(): FilterIsAnyNotNullOrEmpty<T> {
    return this.filterIsAnyNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullAndNotEmpty(): FilterIsAnyNotNullAndNotEmpty<T> {
    return this.filterIsAnyNotNullAndNotEmpty()
}

public fun <T> Iterator<T>.filterIsAny(): FilterIsAny<T> {
    return this.filterIsAny()
}

public fun <T> Iterator<T>.filterIsAnyNullOrEmpty(): FilterIsAnyNullOrEmpty<T> {
    return this.filterIsAnyNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotEmpty(): FilterIsAnyNotEmpty<T> {
    return this.filterIsAnyNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullOrEmpty(): FilterIsAnyNotNullOrEmpty<T> {
    return this.filterIsAnyNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullAndNotEmpty(): FilterIsAnyNotNullAndNotEmpty<T> {
    return this.filterIsAnyNotNullAndNotEmpty()
}

public fun <T> Iterator<T>.filterIsAny(): FilterIsAny<T> {
    return this.filterIsAny()
}

public fun <T> Iterator<T>.filterIsAnyNullOrEmpty(): FilterIsAnyNullOrEmpty<T> {
    return this.filterIsAnyNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotEmpty(): FilterIsAnyNotEmpty<T> {
    return this.filterIsAnyNotEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullOrEmpty(): FilterIsAnyNotNullOrEmpty<T> {
    return this.filterIsAnyNotNullOrEmpty()
}

public fun <T> Iterator<T>.filterIsAnyNotNullAndNotEmpty(): FilterIsAnyNotNullAndNotEmpty<T> {
    return this.filterIsAnyNotNullAndNotEmpty()
Open Source Used In DNAC Platform Arica 13172

```kotlin
Collection) { return when (size) { 0 -> emptySet() 1 -> setOf(if (this is List) this[0] else iterator().next()) else -> toCollection(LinkedHashSet<T>(mapCapacity(size))) } return toCollection(LinkedHashSet<T>(transform).optimizeReadOnlySet()).toList() } * * * * 

Iterable<T>.flatMap(transform: (T) -> Iterable<R>): Iterable<R> { return flatMapTo(ArrayList<R>(transform), transform) } * * * * 

Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C { for (element in this) { val list = transform(element) destination.addAll(list) } return destination } * * * * 

Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, MutableList<T>> { return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector) } * * * * 

Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K): M { for (element in this) { val key = keySelector(element) val list = destination.getOrPut(key) { ArrayList<T>() } list.add(element) } return destination } * * * * 

Iterable<T>.groupBy(crossinline keySelector: (T) -> K): Grouping<T, K> { return object : Grouping<T, K> { override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator() override fun keyOf(element: T): K = keySelector(element) } * * * * 

Iterable<T>.map(transform: (T) -> R): List<R> { return mapTo(ArrayList<R>(collectionSizeOrDefault(10)), transform) } * * * * 

Iterable<T>.mapIndexed(transform: (Int, T) -> R): List<R> { return 
```
mapIndexedTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)  

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original collection.  
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.  
 * @public inline fun <T, R : Any>  
 * @return mapIndexedNotNullTo(ArrayList<R>(), transform)  
 */
public inline fun <T, R : Any, C : MutableCollection<R>> Iterable<T>.mapIndexedNotNullTo(destination: C, transform: (Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?)  
}

/**
 * Applies the given [transform] function to each element and its index in the original collection and appends only the non-null results to the given [destination].  
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.  
 * @public inline fun <T, R : Any>  
 */
public inline fun <T, R : Any, C : MutableCollection<R>> Iterable<T>.mapIndexedNotNullTo(destination: C, transform: (Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
        var index = 0
        for (item in this)
            destination.add(transform(checkIndexOverflow(index++), item))
        return destination
    }
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original collection and appends only the non-null results to the given [destination].  
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.  
 * @public inline fun <T, R : Any>  
 */
public inline fun <T, R : Any, C : MutableCollection<R>> Iterable<T>.mapIndexedNotNullTo(destination: C, transform: (Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
        var index = 0
        for (item in this)
            destination.add(transform(checkIndexOverflow(index++), item))
        return destination
    }
}
is Collection<T> -> LinkedHashSet(this)
else -> toCollection(LinkedHashSet<T>(this)).

*/

/**
 * Returns a set containing all distinct elements from both collections.
 * To get a set containing all elements that are contained in both collections use [intersect].
 */
public infix fun <T> Iterable<T>.union(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet().set.addAll(other)
    // Returns `true` if all elements match the given [predicate].
    return set
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun <T> Iterable<T>.all(predicate: (T) -> Boolean): Boolean {
    // Returns `true` if at least one element matches the given [predicate].
    return iterator().hasNext()
}

/**
 * Returns `true` if collection has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun <T> Iterable<T>.any(): Boolean {
    // Returns the number of elements matching the given [predicate].
    return count(predicate)
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <T> Iterable<T>.anyWithPredicate: Boolean {
    // Returns the number of elements in this collection.
    return count()
}

/**
 * Returns the number of elements in this collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.count(): Int {
    return size
}

/**
 * Returns the number of elements matching the given [predicate].
 */
public inline fun <T> Iterable<T>.count(predicate: (T) -> Boolean): Int {
    // Returns the number of elements in this collection.
    return count()
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <T, R> Iterable<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.
 */
public inline fun <T, R> List<T>.foldRight(initial: R, operation: (T, acc: R) -> R): R {
    var accumulator = initial
    if (!isEmpty()) {
        val iterator = listIterator(size)
        while (iterator.hasPrevious()) {
            accumulator = operation(iterator.previous(), accumulator)
        }
    }
    return accumulator
}

/**
 * Performs the given [action] on each element.
 * 
 * @param [action] function that takes the index of an element, current accumulator value
 * and the element itself, and calculates the next accumulator value.
 */
public inline fun <T, R> List<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

/**
 * Performs the given [action] on each element, providing sequential index with the element.
 * 
 * @param [action] function that takes the index of an element and the element itself
 * and performs the desired action on the element.
 */
public inline fun <T, R> List<T>.forEachIndexed(action: (Int, T) -> Unit): Unit {
    for (index in 0 until size) action(index, list.get(index))
}

/**
 * Returns the largest element or `null` if there are no elements.
 * 
 * If any of elements is NaN returns NaN.
 */
@SinceKotlin("1.1")
public fun
Open Source Used In DNAC Platform Arica  13175

```kotlin
/**
 * Performs the
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public fun <T> Iterable<T>.none(): Boolean {
    if (this is Collection) return isEmpty()
    return !iterator().hasNext()
}

/**
 * Returns `true` if no elements match the
 * 
 * @sample samples.collections.Collections.Aggregates.minBy
 */
public fun <T> Iterable<T>.min(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min == e)
            return e
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the largest element or *null* if there are no elements.
 * 
 * If any of elements is `NaN` returns `NaN`.
 * 
 * If any of elements is *NaN* returns *NaN*.
 * 
 * If any of elements is *NaN* returns *NaN*.
 * 
 * @sample samples.collections.Collections.Aggregates.max
 */
@SinceKotlin("1.1")
public fun Iterable<Float>.max(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or 'null' if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.maxBy
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.max(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max == e)
            return e
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the smallest element or 'null' if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.min
 */
@SinceKotlin("1.1")
public fun Iterable<Float>.min(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min == e)
            return e
        if (min < e) min = e
    }
    return min
}

/**
 * Returns the smallest element or *null* if there are no elements.
 * 
 * If any of elements is *NaN* returns *NaN*.
 * 
 * If any of elements is *NaN* returns *NaN*.
 * 
 * @sample samples.collections.Collections.Aggregates.minBy
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.min(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min == e)
            return e
        if (min < e) min = e
    }
    return min
}
```
inline fun <T, C : Iterable<T>> C.onEach(action: (T) -> Unit): C {
    return apply { for (element in this)
        action(element) }
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.
 * @param operation function that takes the index of an element, current accumulator value and the element itself and calculates the next accumulator value.
 * @param initial the initial accumulator value.
 * @param max the maximum size for the result collection.
 * @throws IllegalArgumentException if the size of the result collection exceeds the maximum.
 * @return a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value.
 * Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value.
 * @param operation function that takes current accumulator value and each element and calculates the next accumulator value.
 * @param initial the initial accumulator value.
 * @param max the maximum size for the result list.
 * @throws IllegalArgumentException if the size of the result list exceeds the maximum.
 * @return a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value.
 * Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> List<T>.reduceRightOrNull(operation: (T, acc: S) -> S): S? {
    val iterator = listIterator(size)
    if (!iterator.hasPrevious())
        throw UnsupportedOperationException("Empty list can't be reduced.")
    var accumulator: S = iterator.previous()
    while (iterator.hasPrevious()) {
        accumulator = operation(iterator.previous(), accumulator)
    }
    return accumulator
}
samples.collections.Collections.Aggregates.scan

*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi

public inline fun <T, R> Iterable<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    \n    var estimatedSize = collectionSizeOrDefault(9)
    \n    if (estimatedSize == 0) return listOf(initial)
    \n    var result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }
    \n    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original collection and current accumulator value that starts with [initial].

*\n* Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

public inline fun <T, R> Iterable<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    \n    val estimatedSize = collectionSizeOrDefault(9)
    \n    if (estimatedSize == 0) return listOf(initial)
    \n    val result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }
    \n    var index = 0
    var accumulator = initial
    for (element in this) {
        accumulator = operation(index++, accumulator, element)
        result.add(accumulator)
    }
    return result
}

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this collection.

*\n* Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

public inline fun <S, T : S> Iterable<T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
    \n    val iterator = this.iterator()
    \n    if (!iterator.hasNext()) return emptyList()
    \n    var accumulator: S = iterator.next()
    \n    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    \n    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original collection and current accumulator value that starts with the first element of this collection.

*\n* Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

public inline fun <S, T : S> Iterable<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    \n    val iterator = this.iterator()
    \n    if (!iterator.hasNext()) return emptyList()
    \n    var accumulator: S = iterator.next()
    \n    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    \n    var index = 1
    while (iterator.hasNext()) {
        accumulator = operation(index++, accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}

Returns an original collection containing all the non-'null' elements, throwing an [IllegalArgumentException] if there are any 'null' elements.

public fun <T : Any> Iterable<T?>.requireNoNulls(): Iterable<T> {
    \n    \n    for (element in this) {
        \n        if (element == null) {
            \n            throw IllegalArgumentException("null element found in $this.")
        }
    }
    \n    \n    return this as Iterable<T>
}

Returns an original collection containing all the non-'null' elements, throwing an [IllegalArgumentException] if there are any 'null'
elements.

```kotlin
    val result = ArrayList<T>(collectionSizeOrDefault(10))
    var removed = false
    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }\n
    @kotlin.internal.InlineOnly
    public inline fun <T> List<T>.minusElement(element: T): List<T> {
        return this.filterNot { it == element }
    }

    @SinceKotlin("1.2")
    public fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R> {
        return windowed(size, size, partialWindows = true, transform = transform)
    }

    @kotlin.internal.InlineOnly
    public inline fun <T> List<T>.chunked(size: Int): List<List<T>> {
        return windowed(size, size, partialWindows = true)
    }

    @SinceKotlin("1.2")
    public fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
        val first = ArrayList<T>()
        val second = ArrayList<T>()
        for (element in this) {
            if (predicate(element)) {
                first.add(element)
            } else {
                second.add(element)
            }
        }
        return Pair(first, second)
    }

    public fun <T> List<T>.plus(element: T): List<T> {
        if (this is Collection) return this.plus(element)
        val result = ArrayList<T>()
        result.addAll(this)
        result.add(element)
        return result
    }

    public operator fun <T> List<T>.minus(element: T): List<T> {
        val result = ArrayList<T>(collectionSizeOrDefault(10))
        var removed = false
        return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }
    }

    public operator fun <T> List<T>.plus(element: T): List<T> {
        if (this is Collection) return this.plus(element)
        val result = ArrayList<T>()
        result.addAll(this)
        result.add(element)
        return result
    }
```

---

Open Source Used In DNAC Platform Arica 13178
Collection<T>.plus(element: T): List<T> {
    val result = ArrayList<T>(size + 1)
    result.add(element)
    return result
}

/**
 * Returns a list containing all elements of the original
 * collection and then all elements of the given [elements] array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)
}

/**
 * Returns a list containing all elements of the original collection and then elements of the given [elements] collection.
 */
@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (windowSize < size && !partialWindows) break
            result.add(List(windowSize) { this[it + index] })
            index += step
        }
        return result
    }
    val result = ArrayList<List<T>>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
        result.add(it)
    }
    return result
}

/**
 * Returns a list of snapshots of the window of the given [size] sliding along this collection with the given [step], where each snapshot is a list.
 * Several last lists may have less elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any
 * @param size * by default "false" which means partial windows won't be preserved
 * @param sample
 * @param samples.collections.Sequences.Transforms
 * @param takeWindows*
 * @param SinceKotlin("1.2")
 */
@SinceKotlin("1.2")
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (windowSize < size && !partialWindows) break
            result.add(List(windowSize) { this[it + index] })
            index += step
        }
        return result
    }
    val result = ArrayList<List<T>>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
        result.add(it)
    }
    return result
}
```kotlin
/*
 * by default `false` which means partial windows won't be preserved
 * \* \* @sample
samples.collections.Sequences.Transformations.averageWindows
 * \*/
@SinceKotlin("1.2")
@public fun <T, R> Iterable<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R):
List<R> { \n    checkWindowSizeStep(size, step) \n    if (this is RandomAccess && this is List) { \n        val thisSize = this.size \n        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1 \n        val result = ArrayList<R>(resultCapacity) \n        val window = MovingSubList(this) \n        var index = 0 \n        while (index in 0 until thisSize) { \n            val windowSize = size.coerceAtMost(thisSize - index) \n            if (!partialWindows && windowSize < size) break \n            window.move(index, index + windowSize) \n            result.add(transform(window)) \n            index += step \n        } \n        return result \n    } \n    val result = ArrayList<R>() \n    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach { \n        result.add(transform(it)) \n    } \n    return result
}

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] array with the same
 * index: \n * \* @sample samples.collections.Iterables.Operations.zip \*/
public infix fun <T, R> Iterable<T>.zip(other: Array<out R>): List<Pair<T, R>> { \n    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` collection and the [other] array with the same
 * index: \n * \* @sample samples.collections.Iterables.Operations.zipWithTransform \*/
public inline fun <T, R, V> Iterable<T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> { \n    val arraySize = other.size \n    val list = ArrayList<V>(minOf(collectionSizeOrDefault(10), arraySize)) \n    var i = 0 \n    for (element in this) { \n        if (i >= arraySize) break \n        list.add(transform(element, other[i++])) \n    } \n    return list
}

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] collection with the same
 * index: \n * \* @sample samples.collections.Iterables.Operations.zipIterable \*/
public infix fun <T, R> Iterable<T>.zip(other: Iterable<R>): List<Pair<T, R>> { \n    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` collection and the [other] collection
 * using the provided [transform] function applied to each pair of elements: \n * \* @sample samples.collections.Iterables.Operations.zipIterableWithTransform \*/
public inline fun <T, R, V> Iterable<T>.zipWithNext(transform: (a: T, b: T) -> V): List<R> { \n    val iterator = iterator() \n    if (!iterator.hasNext()) return emptyList() \n    val result = mutableListOf<R>() \n    var current = iterator.next() \n    while (iterator.hasNext()) { \n        val next = iterator.next() \n        result.add(transform(current, next)) \n        current = next \n    } \n    return result
}
```

buffer.append(prefix)
var count = 0
for (element in this) {
    if (++count > 1) buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) {
            buffer.appendElement(element, transform)
        } else break
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] * elements will be appended, followed by the [truncated] string (which defaults to "...").
 */
public fun <T> Iterable<T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/**
 * Returns this collection as an [Iterable].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.asIterable(): Iterable<T> {
    return this
}

/**
 * Creates a [Sequence] instance that wraps the original collection returning its elements when being iterated.
 */
public fun <T> Iterable<T>.asSequence(): Sequence<T> {
    return Sequence { this.iterator() }
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Iterable<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Iterable<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns a sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}
= 0.0
for (element in this) { \n    sum += element
} \nreturn sum

/**
 * Copyright 2010-2018
 * JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.comparisons.naturalOrder
import kotlin.random.Random

/**
 * Returns the array if it's not `null`, or an empty array otherwise.
 */
@kotlin.internal.InlineOnly
public actual inline fun <T> Array<T>?.orEmpty(): Array<T> = this ?: emptyArray<T>()

@kotlin.internal.InlineOnly
public actual inline fun <T> Collection<T>.toTypedArray(): Array<T> = copyToArray(this)

@JsName("copyToArray")
@PublishedApi
internal fun <T> copyToArray(collection: Collection<*>): Array<T> = if (collection.asDynamic().toArray !== undefined)
    collection.asDynamic().toArray().unsafeCast<Array<T>>()
else
    copyToArrayImpl(collection).unsafeCast<Array<T>>()

@JsName("copyToArrayImpl")
internal actual fun <T> copyToArrayImpl(collection: Collection<*>): Array<T> = copyToArrayImpl(collection).unsafeCast<Array<T>>()

/**
 * Returns an immutable list containing only the specified object [element].
 */
public fun <T> listOf(element: T): List<T> = arrayListOf(element)

/**
 * Returns an immutable set containing only the specified object [element].
 */
public fun <T> setOf(element: T): Set<T> = hashSetOf(element)

/**
 * Returns an immutable map, mapping only the specified key to the [value] specified value.
 */
public fun <K, V> mapOf(pair: Pair<K, V>): Map<K, V> = mapOf(pair)

private fun <T> sortArrayWith(array: Array<T>, comparator: Comparator<in T>) {
    if (array.size <= 1) return

    val sortedArray = copyToArray(array)
    sortArrayWith(sortedArray, comparator)

    for (i in 0 until array.size) {
        array[i] = sortedArray[i]
    }
}


@SinceKotlin("1.3")
@PublishedApi
@JsName("arrayCopy")
internal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    AbstractList.checkRangeIndexes(startIndex, endIndex, source.size)
    val rangeSize = endIndex - startIndex
    AbstractList.checkRangeIndexes(destinationOffset, destinationOffset + rangeSize, destination.size)
    if (js("ArrayBuffer").isView(source) && js("ArrayBuffer").isView(destination)) {
        val subrange = source.asDynamic().subarray(startIndex, endIndex)
        destination.asDynamic().set(subrange, destinationOffset)
    } else {
        if (source !== destination) {
            for (index in rangeSize - 1 downTo 0) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        } else {
            for (index in rangeSize - 1 downTo 0) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        }
    }
}

// Open Source Used In DNAC Platform Arica  13182
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns a [List] containing all key-value pairs.
 * @public
 * @function
 */
public fun <K, V> Map<out K, V>.toList(): List<Pair<K, V>> {
    if (size == 0)
        return emptyList()
    val iterator = entries.iterator()
    if (!iterator.hasNext())
        return emptyList()
    val first = iterator.next()
    if (!iterator.hasNext())
        return listOf(first.toPair())
    val result = ArrayList<Pair<K, V>>(size)
    result.add(first.toPair())
    do {
        result.add(iterator.next().toPair())
    } while (iterator.hasNext())
    return result
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked
 * on each entry of original map.
 * @sample
 * @function
 */
public fun <K, V, R> Map<out K, V>.flatMap(transform: (Map.Entry<K, V>) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Appends all elements yielded from results of [transform] function being invoked on each entry
 * of original map, to the given [destination].
 * @sample
 * @function
 */
public inline fun <K, V, R> Map<out K, V>.flatMapTo(destination: C, transform: (Map.Entry<K, V>) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function to each entry
 * in the original map.
 * @sample
 * @function
 */
public inline fun <K, V, R> Map<out K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function
 * to each entry in the original map.
 * @sample
 * @function
 */
public inline fun <K, V, R : Any> Map<out K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

/**
 * Performs the given [action] on each entry.
 * @sample
 * @function
 */
@kotlin.internal.HidesMembers
public inline fun <K, V> Map<out K, V>.forEach(action: (Map.Entry<K, V>) -> Unit): Unit {
    for (element in this) action(element)
}

/**
 * Returns the number of entries in this map.
 * @sample
 * @function
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.count(): Int {
    return size
}

/**
 * Returns the number of entries matching the given [predicate].
 * @sample
 * @function
 */
public inline fun <K, V> Map<out K, V>.count(predicate: (Map.Entry<K, V>) -> Boolean): Int {
    return count { predicate(it) } + count { !predicate(it) } + count { sample.predicate(it) }
}

/**
 * Returns the first entry yielding the largest value of the given function or `null` if there are
 * no entries.
 * @sample
 * @function
 */
@kotlin.internal.InlineOnly
    return maxBy { properties }()
}

/**
 * Returns the first entry having the largest value according to the provided [comparator] or `null` if there are no entries.
 * @sample
 * @function
 */
@kotlin.internal.InlineOnly
    return maxWith { comparator }()
}
/**
 * Returns the entry yielding the smallest value of the given function
 * or `null` if there are no entries.
 */
    return entries.minBy(selector)
}

/**
 * Returns the entry having the smallest value according to the provided 
 * comparator or `null` if there are no entries.
 */
    return entries.minWith(comparator)
}

/**
 * Returns `true` if the map has no entries.
 */
public fun <K, V> Map<K, V>.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no entries match the given predicate.
 */
public inline fun <K, V> Map<K, V>.none(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the action on each entry and returns the map itself afterwards.
 */
@SinceKotlin("1.1")
public inline fun <K, V, M : Map<K, V>> M.onEach(action: (Map.Entry<K, V>) -> Unit): M {
    return apply { for (element in this) action(element) }
}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>.asIterable(): Iterable<Map.Entry<K, V>> {
    return entries
}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries 
 * when being iterated.
 */
public fun <K, V> Map<K, V>.asSequence(): Sequence<Map.Entry<K, V>> {
    return entries.asSequence()
}
Sequence<T>.first(): T {\n    val iterator = iterator()
    try {
        val element = iterator.next()
        return element
    } catch (e: NoSuchElementException) {
        throw e
    }
}

/**
 * Returns the first element
 * @throws NoSuchElementException if no such element was found.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.first(): T?
{
    var element: T? = null
    while (iterator.hasNext())
        element = iterator.next()
    return element
}

/**
 * Returns the first element matching the given [predicate].
 * @throws NoSuchElementException if no such element was found.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.first(predicate: (T) -> Boolean): T
{
    var element: T? = null
    while (iterator.hasNext())
        if (predicate(iterator.next()))
            element = iterator.next()
    return element
}

/**
 * Returns the first index of [element], or `-1` if the sequence does not contain such element.
 * The operation is _terminal_.
 */
public fun <@kotlin.internal.OnlyInputTypes T>_sequenceOf(element: T): Int
{
    var index = 0
    while (iterator.hasNext())
        if (element == iterator.next())
            return index
        index++
    return -1
}

/**
 * Returns the last index of [element], or `-1` if the sequence does not contain such element.
 * The operation is _terminal_.
 */
public fun <@kotlin.internal.OnlyInputTypes T>_sequenceOfLast(element: T): Int
{
    var index = 0
    while (iterator.hasNext())
        if (element == iterator.next())
            index++
    return -1
}

/**
 * Checks if the sequence is empty.
 */
public fun <T> Sequence<T>.isEmpty(): Boolean
{
    return iterator.hasNext()
}

/**
 * Checks if the sequence is empty or has more than one element.
 */
public fun <T> Sequence<T>.hasOneElement(): Boolean
{
    return iterator.hasNext() && iterator.next() != iterator.next()
}

/**
 * Checks if the sequence contains no element matching the predicate.
 */
public fun <T> Sequence<T>.containsNoElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return !hasOneElement() || iterator.hasNext() && !predicate(iterator.next())
}

/**
 * Checks if the sequence contains an element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return hasOneElement() && predicate(iterator.next())
}

/**
 * Checks if the sequence contains at least one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtLeastOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return iterator.hasNext() || predicate(iterator.next())
}

/**
 * Checks if the sequence contains all elements matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAllElementsMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    while (iterator.hasNext())
        if (!predicate(iterator.next()))
            return false
    return true
}

/**
 * Checks if the sequence contains at most one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtMostOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    for (element in iterator)
        if (predicate(element))
            if (found)
                return false
            found = true
    return found
}

/**
 * Checks if the sequence contains exactly one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsExactlyOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains no element matching the predicate.
 */
public fun <T> Sequence<T>.containsNoElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return !hasOneElement() || iterator.hasNext() && !predicate(iterator.next())
}

/**
 * Checks if the sequence contains an element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return hasOneElement() && predicate(iterator.next())
}

/**
 * Checks if the sequence contains at least one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtLeastOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return iterator.hasNext() || predicate(iterator.next())
}

/**
 * Checks if the sequence contains all elements matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAllElementsMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    while (iterator.hasNext())
        if (!predicate(iterator.next()))
            return false
    return true
}

/**
 * Checks if the sequence contains at most one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtMostOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains exactly one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsExactlyOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains no element matching the predicate.
 */
public fun <T> Sequence<T>.containsNoElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return !hasOneElement() || iterator.hasNext() && !predicate(iterator.next())
}

/**
 * Checks if the sequence contains an element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return hasOneElement() && predicate(iterator.next())
}

/**
 * Checks if the sequence contains at least one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtLeastOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return iterator.hasNext() || predicate(iterator.next())
}

/**
 * Checks if the sequence contains all elements matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAllElementsMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    while (iterator.hasNext())
        if (!predicate(iterator.next()))
            return false
    return true
}

/**
 * Checks if the sequence contains at most one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtMostOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains exactly one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsExactlyOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains no element matching the predicate.
 */
public fun <T> Sequence<T>.containsNoElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return !hasOneElement() || iterator.hasNext() && !predicate(iterator.next())
}

/**
 * Checks if the sequence contains an element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return hasOneElement() && predicate(iterator.next())
}

/**
 * Checks if the sequence contains at least one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtLeastOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    return iterator.hasNext() || predicate(iterator.next())
}

/**
 * Checks if the sequence contains all elements matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAllElementsMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    while (iterator.hasNext())
        if (!predicate(iterator.next()))
            return false
    return true
}

/**
 * Checks if the sequence contains at most one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsAtMostOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
                return false
            lastFound = found
            found = true
    return found
}

/**
 * Checks if the sequence contains exactly one element matching the given [predicate].
 */
public fun <T> Sequence<T>.containsExactlyOneElementMatchingPredicate(predicate: (T) -> Boolean): Boolean
{
    var found = false
    var lastFound = false
    for (element in iterator)
        if (predicate(element))
            if (lastFound)
Boolean): T

    val single: T? = null
    var found = false

    for element in this
    {
        if (predicate(element))
        {
            if (found) throw IllegalArgumentException("Sequence contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Sequence contains no element matching the predicate.")

    @Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns single element, or `null` if the sequence is empty or has more than one element.
 *
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.singleOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    val single = iterator.next()
    if (iterator.hasNext())
        return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
 *
 * The operation is _terminal_ and _public inline fun_<T> Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T?

    val single: T? = null
    var found = false

    for (element in this)
    {
        if (predicate(element))
        {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a sequence containing all elements except first [n] elements.
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.drop(n: Int): Sequence<T> {
    require(n >= 0) { "Requested element count \$n is less than zero." }
    return when {
        n == 0 -> this
        this is DropTakeSequence -> this.drop(n)
        else -> DropSequence(this, n)
    }
}

/**
 * Returns a sequence containing all elements except first elements that satisfy the given [predicate].
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.dropWhile(predicate: (T) -> Boolean): Sequence<T> {
    return DropWhileSequence(this, predicate)
}

/**
 * Returns a sequence containing only elements matching the given [predicate].
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.filter(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, true, predicate)
}

/**
 * Returns a sequence containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed {
        index, element ->
            if (predicate(index, element)) destination.add(element)
    }
    return destination
}

/**
 * Returns a sequence containing all elements instances of specified type parameter R.
 *
 * The operation is _intermediate_ and _stateless_.
 */
public inline fun <reified R> Sequence<*>.filterIsInstance(): Sequence<@kotlin.internal.NoInfer R> {
    @Suppress("UNCHECKED_CAST")
    return filter { it is R } as Sequence<R>
}

/**
 * Appends all elements that are instances of specified type parameter R to the given [destination].
 *
 * The operation is _intermediate_ and _stateless_.
 */
public inline fun <reified R, C : MutableCollection<in R>> Sequence<*>.filterIsInstanceTo(destination: C) {
    for (element in this) if (element is R) destination.add(element)
}

/**
 * Returns a sequence containing all elements that are not matching the given [predicate].
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.filterNot(predicate: (T) -> Boolean): Sequence<T> {
    return FilteringSequence(this, false, predicate)
}

/**
 * Returns a sequence containing all elements that are not `null`.
 *
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T : Any> Sequence<T?>.filterNotOrNull(): Sequence<T?> {
    @Suppress("UNCHECKED_CAST")
    return filterNot { it == null } as
Sequence<T>

/**
 * Appends all elements that are not `null` to the given [destination].
 */
public fun <C : MutableCollection<in T>, T : Any> Sequence<T>.filterNotNullTo(destination: C): C {
    C {}
    for (element in this) if (element != null) destination.add(element)
    return destination
}

/**
 * Appends all elements not matching the given [predicate] to the given [destination].
 */
public fun <C :MutableCollection<in T>, T: Any> Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    C {}
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
public fun <C :MutableCollection<in T>, T: Any> Sequence<T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    C {}
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a sequence containing first [n] elements.
 */
public fun <T> Sequence<T>.take(n: Int): Sequence<T> {
    val sortedList = this@take.take(n)
    return sortedList
}

/**
 * Returns a sequence containing first elements satisfying the given [predicate].
 */
public fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {
    return TakeWhileSequence(this, predicate)
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.
 */
public fun <T : Comparable<T>> Sequence<T>.sorted(): Sequence<T> {
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this@sorted.toMutableList()
            sortedList.sort()
            return sortedList.iterator()
        }
    }
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to natural sort order of the value returned by specified [selector] function.
 */
public fun <T, R : Comparable<R>> Sequence<T>.sortedBy(selector: (T) -> R?): Sequence<T> {
    return sortedWith(compareBy(selector))
}

/**
 * Returns a sequence that yields elements of this sequence sorted descending according to their natural sort order.
 */
public fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator].
 */
public fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T> {
    return object : Sequence<T> {
        override fun iterator(): Iterator<T> {
            val sortedList = this@sortedWith.toMutableList()
            sortedList.sortWith(comparator)
            return sortedList.iterator()
        }
    }
}

/**
 * Returns a [Map] containing key-value pairs provided by [transform] function applied to elements of the given sequence.
 */
public fun <K, V> Sequence<in T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val sortedList = sortedWith(LinkedHashMap<K, V>())
    return sortedList
}

Open Source Used In DNAC Platform Arica 13189
key
* returned from [keySelector] function applied to each element.\n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* The returned map preserves the entry iteration order of the original sequence.\n* The operation is _terminal_.\n* @sample
samples.collections.Collections(Transformations.associateBy)/n*/npublic inline fun <T, K>
Sequence<T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n    return associateByTo(LinkedHashMap<K, T>(), keySelector)\n}\n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given sequence.\n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* The returned map preserves the entry iteration order of the original sequence.\n* The operation is _terminal_.\n* @sample
samples.collections.Collections(Transformations.associateByWithValueTransform)/n*/npublic inline fun <T, K, V>
Sequence<T>.associateByWithValueTransform(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, V> {\n    return associateByTo(LinkedHashMap<K, V>(), keySelector, valueTransform)\n}\n/**
 * Populates and returns the [destination] mutable map with key-value pairs,\n* where key is provided by the [keySelector] function applied to each element of the given sequence\n* and value is the element itself.\n* If any two elements would have the same key returned by [keySelector] the last one gets added to the map.\n* The operation is _terminal_.\n* @sample
samples.collections.Collections(Transformations.associateByTo)/n*/npublic inline fun <T, K, M :
MutableMap&lt;K, in T&gt;>
Sequence&lt;T&gt;.associateByTo(destination: M, keySelector: (T) -> K): M {\n    for (element in this) {\n        destination.put(keySelector(element), element)\n    }\n    return destination\n}\n/**
 * Returns a [HashSet] of all elements.\n* The operation is _terminal_.\n* @sample
samples.collections.Collections(Transformations.associateWith)/n*/npublic inline fun <K, V>
Sequence&lt;K&gt;.associateWith(valueSelector: (K) -> V): HashSet&lt;V&gt; {\n    val result = LinkedHashSet&lt;V&gt;()\n    return associateWithTo(result, valueSelector)\n}\n*/n
* Returns a [HashSet] of all elements.\n* The operation is _terminal_.\n* @sample
samples.collections.Collections(Transformations.associateWithTo)/n*/npublic inline fun <T, C :
MutableCollection&lt;in T&gt;>
Sequence&lt;T&gt;.toCollection(destination: C): C {\n    for (item in this) {\n        destination.add(item)\n    }\n    return destination\n}\n*/n
* Returns a [HashSet] of all elements.\n* The operation is _terminal_.\n* @sample
samples.Collections(Transformations.toCollection)/n*/n
return toCollection(HashSet<T>())
}

/**
 * Returns a [List] containing all elements.
 *
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toList(): List<T> {
    return this.toMutableList().optimizeReadOnlyList()
}

/**
 * Returns a [MutableList] filled with all elements of this
sequence.
 *
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toMutableList(): MutableList<T> {
    return toCollection(ArrayList<T>())
}

/**
 * Returns a [Set] of all elements.
 *
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toSet(): Set<T> {
    return toCollection(LinkedHashSet<T>()).optimizeReadOnlySet()
}

/**
 * Returns a single sequence of all elements
from results of [transform] function being invoked on each element of original sequence.
 *
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Transformations.flatMap
 */
public fun <T, R> Sequence<T>.flatMap(transform: (T) -> Sequence<R>): Sequence<R> {
    return FlatteningSequence(this, transform, { it.iterator() })
}

/**
 * Appends all elements yielded from results of
[transform] function being invoked on each element of original sequence, to the given [destination].
 *
 * The operation is _terminal_.
 */
public inline fun <T, R, C : MutableCollection<in R>>
    Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of
the original sequence by the key returned by the given [keySelector] function
applied to each element and
returns a map where each group key is associated with a list of corresponding elements.
 *
 * The returned map preserves the entry iteration order of the keys produced from the original sequence.
 *
 * The operation is _terminal_.
 */
public inline fun <T, K>
    Sequence<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function
applied to each element of the original sequence
by the key returned by the given [keySelector] function applied to the element
and returns a map where each group key is associated with a list of corresponding values.
 *
 * The operation is _terminal_.
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>>
    Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold
operations
using the specified [keySelector] function to extract a key from each element.
 *
 * The operation is _intermediate_ and _stateless_.
 * 
 * @sample samples.collections.Transformations.groupingByEachCount
 */
@SinceKotlin("1.1")
public inline fun <T, K, V>
    Sequence<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V):
        Map<K, List<V>> {
        return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
    }

/**
 * Appends all elements yielded from results of
[transform] function being invoked on each element of original sequence, to the given [destination].
 *
 * The operation is _terminal_.
 * 
 * @sample samples.collections.Transformations.groupByKeysAndValues
 */
public inline fun <T, K, V, M : MutableMap<in K, MutableList<V>>>
    Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList() }
        list.add(valueTransform(element))
    }
    return destination
}

/**
 * Groups elements of
the original sequence by the key returned by the given [keySelector] function
applied to each element and
returns a map where each group key is associated with a list of corresponding elements.
 *
 * The operation is _terminal_.
 */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>>
    Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList() }
        list.add(element)
    }
    return destination
}
this@groupingBy.iterator()@ override fun keyOf(element: T): K = keySelector(element)@

* The operation is _intermediate_ and _stateless_, @sample

Open Source Used In DNAC Platform Arica  13192

samples.Collections.Collections.Transformations.map(transform: (T) -> R): Sequence<R> { return TransformingSequence(this, transform) returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> { return TransformingSequence(this, transform) returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.mapIndexed(transform: (Int, T) -> R): Sequence<R> { return TransformingIndexedSequence(this, transform) returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.mapIndexedNotNull(transform: (Int, T) -> R?): Sequence<R> { return TransformingIndexedSequence(this, transform).filterNotNull() returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (Int, T) -> R?): C { var index = 0

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.mapIndexedTo(destination: C, transform: (Int, T) -> R): C { return TransformingIndexedSequence(this, transform).filterNotNull() returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, R> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C { destination.add(transform(item))

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T> Sequence<T>.withIndex(): Sequence<IndexedValue<T>> { return IndexingSequence(this)

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T> Sequence<T>.distinctAndDistinctBy(destination: C, transform: (T) -> R): C { return TransformingSequence(this, transform) returns a sequence containing

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T> Sequence<T>.distinct(): Sequence<T> { return this@Sequence<T>.distinct() returns a sequence containing

* having distinct keys returned by the given [selector] function.

* @param [transform] function that takes the index of an element and the element itself

* and returns the result of the transform applied to the element.

* The operation is _intermediate_ and _stateless_, @sample

public fun <T, K>
Sequence<T>.distinctBy(selector: (T) -> K): Sequence<T> { return DistinctSequence(this, selector) } * Returns a mutable set containing all distinct elements from the given sequence. *

Sequence<T>.toMutableSet(): MutableSet<T> { val set = LinkedHashSet<T>() for (item in this) set.add(item) return set } * Returns 'true' if all elements match the given [predicate]. *

Sequence<T>.all(predicate: (T) -> Boolean): Boolean { for (element in this) if (!predicate(element)) return false } * Returns 'true' if there is at least one element. *

Sequence<T>.any(): Boolean { return iterator().hasNext() } * Returns 'true' if at least one element matches the given [predicate]. *

Sequence<T>.any(predicate: (T) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true } * Returns the number of elements in this sequence. *

Sequence<T>.count(): Int { var count = 0 for (element in this) checkCountOverflow(++count) return count } * Returns the number of elements matching the given [predicate]. *

Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R { var accumulator = initial for (element in this) accumulator = operation(accumulator, element) return accumulator } * Returns the largest element or 'null' if there are no elements. *

Sequence<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R { var index = 0 var accumulator = initial for (item in this) accumulator = operation(checkIndexOverflow(index++), accumulator, item) return accumulator } * Returns the largest element or 'null' if there are no elements. *

Sequence<T>.max(): Double? { var iterator = iterator() if (!iterator.hasNext()) return null var max = iterator.next() if (max is NaN) return max } * Returns the largest element or 'null' if there are no elements. *

Sequence<T>.max(): Float? { var iterator = iterator() if (!iterator.hasNext()) return null var max = iterator.next() if (max is NaN) return max } * Returns the largest element or 'null' if there are no elements. *

Sequence<T>.max(): T? { var iterator = iterator() if (!iterator.hasNext()) return null var max = iterator.next() while (iterator.hasNext()) { var e = iterator.next() if (e is NaN) return null var max = iterator.next() if (max is NaN) return max } * Returns the first element yielding the largest value of the given function or 'null' if there are no elements. *

Sequence<T>.any(predicate: (T) -> Boolean): Boolean { for (element in this) if (predicate(element)) return true } * Returns 'true' if all elements match the given [predicate]. *
samples.collections.Collections.Aggregates.maxBy

```kotlin
public inline fun <T, R : Comparable<R>> Sequence<T>.maxBy(selector: (T) -> R): T? {
    var maxElem = iterator.next()!!
    var maxVal = selector(maxElem)!!
    do {
        val e = iterator.next()!!
        val v = selector(e)!!
        if (v > maxVal) {
            maxElem = e
            maxVal = v
        }
    } while (iterator.hasNext())
    return maxElem
}
```

The operation is _terminal_. The operation is _terminal_ and _terminal_.

```kotlin
public fun <T> Sequence<T>.maxWith(comparator: Comparator<T>): T? {
    var min = iterator.next()!!
    while (iterator.hasNext()) {
        val e = iterator.next()!!
        if (comparator.compare(min, e) < 0) min = e
    }
    return min
}
```

The operation is _terminal_ and _terminal_. The operation is _terminal_.

```kotlin
public fun Sequence<Double>.min(): Double? {
    var min = iterator.next()!!
    if (min.isNaN()) return min
    while (iterator.hasNext()) {
        val e = iterator.next()!!
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}
```

The operation is _terminal_ and _terminal_. The operation is _terminal_.

```kotlin
public fun <T> Sequence<T>.none(): Boolean {
    return !iterator().hasNext()
}
```

The operation is _terminal_ and _terminal_. The operation is _terminal_.

```kotlin
public inline fun <T> Sequence<T>.reduce(operation: (S, T) -> S): S {
    var accumulator: S = iterator.next()!!
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next()!!)
    }
    return accumulator
}
```

The operation is _terminal_ and _terminal_. The operation is _terminal_.

```kotlin
public inline fun <T> Sequence<T>.reduce(operation: (S, T) -> S): S {
    var accumulator: S = iterator.next()!!
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next()!!)
    }
    return accumulator
}
```

The operation is _terminal_ and _terminal_. The operation is _terminal_.

```kotlin
public fun <T> Sequence<T>.reduce(operation: (S, T) -> S): S {
    var accumulator: S = iterator.next()!!
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next()!!)
    }
    return accumulator
}
```
[operation] function that takes the index of an element, current accumulator value * and the element itself and calculates the next accumulator value.

* The operation is _terminal_.

* @sample

```kotlin
fun <S, T : S> Sequence<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    var index = 1
    var accumulator: S = iterator.next() while (iterator.hasNext()) {
        accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the sequence is empty.

* The operation is _terminal_.

* @sample

```kotlin
fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next())
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element, and calculates the next accumulator value.

* The operation is _terminal_.

* @sample

```kotlin
fun <T, R> Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var accumulator = initial
        for (element in this) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.

* @sample

```kotlin
fun <T, R> Sequence<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var index = 0
        var accumulator = initial
        for (element in this) {
            accumulator = operation(checkIndexOverflow(index++), accumulator, element)
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with [initial] value. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * The [initial] value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence's lazy nature.

* @sample

```kotlin
fun <S, T : S> Sequence<T>.scanReduce(operation: (acc: S, T) -> S): Sequence<S> {
    return sequence {
        var accumulator: S = iterator.next() while (iterator.hasNext()) {
            accumulator = operation(accumulator, iterator.next())
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with the first element of this sequence. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * @param

```kotlin
fun <T, R> Sequence<T>.scanIndexed(initial: T, operation: (acc: T, R, T) -> T): Sequence<T> {
    return sequence {
        yield(initial)
        var index = 0
        var accumulator = initial
        for (element in this) {
            accumulator = operation(checkIndexOverflow(index++), accumulator, element)
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * @param

```kotlin
fun <T, R> Sequence<T>.scanReduce(operation: (acc: T, R, T) -> T): Sequence<T> {
    return sequence {
        var accumulator: T = iterator.next() while (iterator.hasNext()) {
            accumulator = operation(accumulator, iterator.next())
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * @param

```kotlin
fun <T, R> Sequence<T>.scanIndexed(initial: T, operation: (index: Int, acc: T, R, T) -> T): Sequence<T> {
    return sequence {
        yield(initial)
        var index = 0
        var accumulator = initial
        for (element in this) {
            accumulator = operation(checkIndexOverflow(index++), accumulator, element)
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. * @param

```kotlin
fun <T, R> Sequence<T>.scanReduce(operation: (acc: T, R, T) -> T): Sequence<T> {
    return sequence {
        var accumulator: T = iterator.next() while (iterator.hasNext()) {
            accumulator = operation(accumulator, iterator.next())
            yield(accumulator)
        }
    }
}
```

Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.
with the first element of this sequence.

Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator value, and the element itself, and calculates the next accumulator value.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.

The operation is _intermediate_ and _stateless_. The operation takes the index of an element, current accumulator, and the element itself, and returns the sum of all values produced by [selector] function applied to each element in the sequence.
elements contained in the given [elements] collection.

Note that the source sequence and the collection being subtracted are iterated only when an `iterator` is requested from it. Changing any of them between successive calls to `operator minus` may affect the result. The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations. The operation is _intermediate_ and _stateful_.

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.minusElement(element: T): Sequence<T> {
    return minus(element)
}
```

**Sequence<T> return object:**

```kotlin
Sequence<T> { 
    override fun iterator(): Iterator<T> {
        return object: Sequence<T> {
            override fun iterator(): Iterator<T> {
                val other = elements.convertToSetForSetOperation()
                if (other.isEmpty())
                    return this@minus.iterator()
                else
                    return this@minus.filterNot { it in other }.iterator()
            }
        }
    }
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
            else
                return this@minus.filterNot { it in other }.iterator()
        }
    }
}
```

**Sequence<T>.partition(predicate: (T) -> Boolean):**

```kotlin
partition(predicate) { 
    val first = ArrayList<T>()
    val second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            return this@minus.filterNot { it in other }.iterator()
        }
    }
}
```

**Sequence<T>.plus(element: T):**

```kotlin
plus(element) { 
    return this + sequenceOf(element).flatten()
}
```

**Sequence<T>.plus(elements: Array<out T>):**

```kotlin
Sequence<T> { 
    return this + elements.asList().asSequence().flatten()
}
```

**Sequence<T>.plus(elements: Sequence<T>):**

```kotlin
Sequence<T> { 
    return this + elements.asSequence().flatten()
}
```

**Sequence<T>.plus(elements: Iterable<T>):**

```kotlin
Sequence<T> { 
    return this + sequenceOf(elements).flatten()
}
```

**Sequence<T>.plus(elements: Iterable<T>):**

```kotlin
Sequence<T> { 
    return this + sequenceOf(elements).asSequence().flatten()
}
```

**Sequence<T>.plus(Sequence<T>):**

```kotlin
Sequence<T> { 
    return this + sequenceOf(Sequence(elements).asSequence()).flatten()
}
```

**Sequence<T>.plus(Sequence<T>):**

```kotlin
Sequence<T> { 
    return this + sequenceOf(Sequence(elements).asSequence()).flatten()
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            return this@plus.filterNot { it in other }.iterator()
        }
    }
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            return this@plus.filterNot { it in other }.iterator()
        }
    }
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            return this@plus.filterNot { it in other }.iterator()
        }
    }
}
```

**Sequence<T>:**

```kotlin
Sequence<T> { 
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            return this@plus.filterNot { it in other }.iterator()
        }
    }
}
```
return plus(element)/*@Sample samples.collections.Sequences.Transformations.takeWindows
    return windowedSequence(size, step, partialWindows, reuseBuffer = false)/*@*/
}

/**
 * Returns a sequence of results of applying the given [transform] function to
 * each list representing a view over the given [size] sliding along this sequence with the given [step].
 * 
 * Several last lists may have less elements than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * 
 * Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 * 
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default false
 * which means partial windows won't be preserved
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default false
 * which means partial windows won't be preserved
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 */
public fun <T, R> Sequence<T>.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (List<T>) -> R): Sequence<R> {
    return windowedSequence(size, step, partialWindows, reuseBuffer = true).map(transform)/*@*/
}

/**
 * Returns a sequence of values built from the elements of this sequence and the [other] sequence with the same index.
 * The resulting sequence ends as soon as the shortest input sequence ends.
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 */
infix fun <T, R> Sequence<T>.zip(other: Sequence<R>): Sequence<Pair<T, R>> {
    return MergingSequence(this, other) { t1, t2 -> t1 to t2 }/*@*/
}

/**
 * Returns a sequence containing the results of applying the given [transform] function to each pair of two adjacent elements in this sequence.
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 * 
 * @sinceKotlin("1.2")
 * 
 * The operation is intermediate and stateless.
 */
fun <T, R, V> Sequence<T>.zip(other: Sequence<R>, transform: (a: T, b: R) -> V): Sequence<V> {
    return MergingSequence(this, other, transform)/*@*/
}

/**
 * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * 
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * 
 * The operation is terminal.
 */
public fun <T, A : Appendable>
sequence.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = ", ", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A

{ }
buffer.append(prefix)
var count = 0
for (element in this) {
if (++count > 1)
buffer.append(separator)
if (limit < 0 || count <= limit) { }
buffer.appendElement(element, transform)
}
else break
}
buffer.append(postfix)
return buffer
}

/**
 * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to ", ...,").
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.joinTo(separator: CharSequence = ", ", prefix: CharSequence = ", ", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()}

/**
 * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.
 */
public fun <T> Sequence<T>.asIterable(): Iterable<T> {
return Iterable { this.iterator() }
}

/**
 * Returns this sequence as a [Sequence].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.asSequence(): Sequence<T> {
return this
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Sequence<Byte>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Sequence<Short>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Sequence<Int>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Sequence<Long>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Sequence<Float>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Sequence<Double>.average(): Double {
var sum: Double = 0.0
var count: Int = 0
for (element in this) {
sum += element
checkCountOverflow(++count)
}
return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Sequence<Byte>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Sequence<Short>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Sequence<Int>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Sequence<Long>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Sequence<Float>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}

/**
 * Returns the sum of all elements in the sequence.
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Sequence<Double>.sum(): Int {
var sum: Int = 0
for (element in this) {
sum += element
return sum
}
}
Sequence<Long>.sum(): Long { var sum: Long = 0L for (element in this) { sum += element
} return sum
}/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Sequence<Float>.sum(): Float { var sum: Float = 0.0f for (element in this) { sum += element
} return sum
}
/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Sequence<Double>.sum(): Double { var sum: Double = 0.0 for (element in this) { sum += element
} return sum
}/**
 * Returns a set containing all elements of the original set except the given [element].
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.minus(element: T): Set<T> {
    var removed = false
    return this.filterTo(result) { if (!removed && it == element) { removed = true; false } else true }
}
/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements].
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    var removed = false
    return this.filterTo(result) { if (!removed && it in elements) { removed = true; false } else true }
}
/**
 * Returns a set containing all elements of the original set except the given [elements].
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    var removed = false
    return this.filterTo(result) { if (!removed && it in elements) { removed = true; false } else true }
}
/**
 * Returns a set containing all elements of the original set except the given [element].
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}
/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(element: T): Set<T> {
    var result = LinkedHashSet<T>(mapCapacity(size + 1))
    result.addAll(this)
    result.add(element)
    return result
}
val result = LinkedHashSet<T>(mapCapacity(elements.collectionSizeOrNull() ?: let { this.size + it } ?: this.size * 2))
result.addAll(this)
result.addAll(elements)
return result

set<T> plus(elements: Sequence<T>): Set<T> {
    return result.addAll(elements)
}

/**
 * Returns a set containing all elements of the original set and the given [elements] sequence, which aren't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    val result = LinkedHashSet<T>(mapCapacity(this.size * 2))
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

import kotlin.random.*

/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns the first character matching the given [predicate], or `null` if no such character was found.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
not contain such character.

*\n
*/

public inline fun CharSequence.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

*\n
*/

public inline fun CharSequence.last(): Char {
    if (isEmpty()) throw NoSuchElementException("Char sequence is empty.")
    return this[lastIndex]
}

*\n
*/

public fun CharSequence.lastOrNull(): Char? {
    return if (isEmpty()) null else this[lastIndex]
}

*\n
*/

public fun CharSequence.random(): Char {
    return random(Random)
}

*\n
*/

public fun CharSequence.random(random: Random): Char {
    if (isEmpty()) throw NoSuchElementException("Char sequence is empty.")
    return get(random.nextInt(length))
}

*\n
*/

public fun CharSequence.single(): Char {
    return when (length) {
        0 -> throw NoSuchElementException("Char sequence is empty.")
        1 -> this[0]
        else -> throw IllegalArgumentException("Char sequence has more than one element.")
    }
}

*\n
*/

public fun CharSequence.singleOrNull(): Char? {
    return if (length == 1) this[0] else null
}

*\n
*/

public inline fun CharSequence.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Char sequence contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

*\n
*/

public fun CharSequence.drop(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(n.coerceAtMost(length), length)
}

*\n
*/

public fun String.drop(n: Int): String {
    require(n >= 0) { "Requested character
count $n$ is less than zero."

    return substring(n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence with the last $n$ characters removed.
 * @throws IllegalArgumentSelectorException if $n$ is negative.
 * @sample samples.text.Strings.drop$(n * @sample samples.text.Strings.drop)n
 */
public fun CharSequence.dropLast(n: Int):
    return substring(n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing all characters except first characters that satisfy the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself and returns the result of predicate evaluation on the character.
 */
public inline fun CharSequence.dropFirstWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return ""
}

/**
 * Returns a string containing all characters except first characters that satisfy the given [predicate].
 */
public inline fun String.dropFirstWhile(predicate: (Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].
 * @param [predicate] function that takes the index of a character and the character itself and returns the result of predicate evaluation on the character.
 */
public inline fun CharSequence.filterIndexed(predicate: (index: Int, Char) -> Boolean): CharSequence {
    return filterIndexedTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing all characters except first characters that satisfy the given [predicate].
 */
public inline fun String.filterIndexed(predicate: (index: Int, Char) -> Boolean): String {
    return filterIndexedTo(StringBuilder(), predicate).toString()
}

/**
 * Appends all characters matching the given [predicate] to the given [destination].
 */
public inline fun <C : Appendable> CharSequence.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {
    forEachIndexed {
        index, element ->
            if (predicate(index, element)) destination.append(element)
    }
    return destination
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that do not match the given [predicate].
 */
public inline fun String.filterNotIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {
    forEachIndexed {
        index, element ->
            if (predicate(index, element)) destination.append(element)
    }
    return destination
}

/**
 * Returns a string containing only those characters from the original string that do not match the given [predicate].
 */
public inline fun String.filterNotIndexed(predicate: (index: Int, Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that do not match the given [predicate].
 */
public inline fun CharSequence.filterNotIndexed(predicate: (Int, Char) -> Boolean): CharSequence {
    return filterNotIndexedTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing only those characters from the original string that do not match the given [predicate].
 */
public inline fun String.filterNotIndexed(predicate: (Int, Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing all characters matching the given [predicate].
 */
public inline fun CharSequence.filterNotIndexedTo(destination: C, predicate: (Int, Char) -> Boolean): C {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}

/**
 * Returns a string containing all characters matching the given [predicate].
 */
public inline fun String.filterNotIndexedTo(destination: String, predicate: (Int, Char) -> Boolean): String {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that match the given [predicate].
 */
public inline fun CharSequence.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}

/**
 * Returns a string containing only those characters from the original string that match the given [predicate].
 */
public inline fun String.filterIndexedTo(destination: String, predicate: (Int, Char) -> Boolean): String {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that are matched by the given [predicate].
 */
public inline fun CharSequence.filterIndexedTo(destination: C, predicate: (Int, Char) -> Boolean): C {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}

/**
 * Returns a string containing only those characters from the original string that are matched by the given [predicate].
 */
public inline fun String.filterIndexedTo(destination: String, predicate: (Int, Char) -> Boolean): String {
    for (index in this.indices)
        if (predicate(index, this[index]))
            destination.append(this[index])
    return destination
}
```kotlin
@kotlin.internal.InlineOnly
public inline fun String.reversed(): String {
    return (this as
        CharSequence).reversed().toString()
}

/**
 * Returns a [Map] containing key-value pairs provided by
 * [transform] function applied to characters of the given char sequence.
 * The returned map preserves the entry iteration order of the original char sequence.
 * If any two pairs would have the same key the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original char sequence.
 * @param<K, V> @sample samples.text.Strings.associate
 */
public inline fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity),
        transform).associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector).associateWithByTo(LinkedHashMap<K, V>(capacity),
                valueSelector)
    return destination.put(keySelector(element), element) return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function and value is provided by the [valueSelector] function applied to characters of the given char sequence.
 * The returned map preserves the entry iteration order of the original char sequence.
 * If any two characters would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original char sequence.
 * @sample samples.text.Strings.associate
 */
public inline fun <V> CharSequence.associateWith(valueSelector: (Char) -> V): Map<Char, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateWithTo(LinkedHashMap<Char, V>(capacity),
        valueSelector).associateWithByTo(LinkedHashMap<Char, V>(capacity),
            valueSelector)
    return destination
}
```

```kotlin
in V>> CharSequence.associateWithTo(destination: M, valueSelector: (Char) -> V): M {\n    for (element in this) {\n        destination.put(element, valueSelector(element))\n    }\n    return destination\n}\n
in */anpublic fun C: MutableCollection<in Char>>
CharSequence.toCollection(destination: C): C {\n    for (item in this) {\n        destination.add(item)\n    }\n    return destination\n}\n
in */anpublic fun CharSequence.toHashSet(): HashSet<Char> {\n    return HashSet<Char>(mapCapacity(length))\n}\n
in */anpublic fun CharSequence.toList(): List<Char> {\n    return when (length) {\n        0 -> emptyList()\n        1 -> listOf(this[0])\n        else -> toMutableList()\n    }\n}\n
in */anpublic fun CharSequence.toMutableList(): MutableList<Char> {\n    return toCollection(ArrayList<Char>(length))\n}\n
in */anpublic fun CharSequence.toSet(): Set<Char> {\n    return when (length) {\n        0 -> emptySet()\n        1 -> setOf(this[0])\n        else -> toCollection(LinkedHashSet<Char>(mapCapacity(length)))\n    }\n}\n
in */anpublic inline fun <R> CharSequence.flatMap(transform: (Char) -> Iterable<R>): List<R> {\n    return flatMapTo(ArrayList<R>(), transform)\n}\n
in */anpublic inline fun <R, C : MutableCollection<in R>> CharSequence.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {\n    for (element in this) {\n        val list = transform(element)\n        destination.addAll(list)\n    }\n    return destination\n}\n
in */anpublic inline fun <K> CharSequence.groupBy(keySelector: (Char) -> K): Map<K, List<Char>> {\n    return groupByTo(LinkedHashMap<K, MutableList<Char>>(), keySelector)\n}\n
in */anpublic inline fun <K, V, M : MutableMap<in K, MutableList<in V>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<Char>() }\n        list.add(element)\n    }\n    return destination\n}\n
in */anpublic inline fun <K, V, M : MutableMap<in K, LinkedList<in V>>> CharSequence.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<Char>() }\n        list.add(valueTransform(element))\n    }\n    return destination\n}\n```

Open Source Used In DNAC Platform Arica 13208

```kotlin
CharSequence.count(): Int { return length }

/**
 * Returns the number of characters matching the given [predicate].
 */
public inline fun CharSequence.count(predicate: (Char) -> Boolean): Int { var count = 0
    for (element in this) if (predicate(element)) ++count
    return count }

/**
 * Accumulates value starting with [initial] and applying [operation] from left to right to current accumulator value and each character.
 */
public inline fun <R> CharSequence.fold(initial: R, operation: (acc: R, Char) -> R): R { var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator }

/**
 * Accumulates value starting with [initial] and applying [operation] from right to left to each character and current accumulator value.
 */
public inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R { var index = lastIndex
    var accumulator = initial
    while (index >= 0) { accumulator = operation(get(index--), accumulator) }
    return accumulator }

/**
 * Performs the given [action] on each character.
 */
public inline fun CharSequence.forEach(action: (Char) -> Unit): Unit { for (element in this) action(element) }

/**
 * Performs the given [action] on each character, providing sequential index with the character.
 */
public inline fun CharSequence.forEachIndexed(action: (index: Int, Char) -> Unit): Unit { var index = 0
    for (item in this) action(index++, item) }

/**
 * Returns the largest character or `null` if there are no characters.
 */
public fun CharSequence.max(): Char? { if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) { val e = this[i]
        if (max < e) max = e
    }
    return max }

/**
 * Returns the first character yielding the largest value of the given function or `null` if there are no characters.
 */
public inline fun <R : Comparable<R>> CharSequence.maxBy(selector: (Char) -> R): Char? { if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) { val e = this[i]
        val v = selector(e)
        if (maxValue < v) { maxElem = e
            maxValue = v
        }
    }
    return maxElem }

/**
 * Returns the smallest character or `null` if there are no characters.
 */
public fun CharSequence.min(): Char? { if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { val e = this[i]
        if (min > e) min = e
    }
    return min }

/**
 * Returns the first character having the smallest value according to the provided [comparator] or `null` if there are no characters.
 */
public fun CharSequence.minBy(comparator: Comparator<in Char>): Char? { if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return min }

/**
 * Returns the smallest character or `null` if there are no characters.
 */
public fun CharSequence.minWith(comparator: Comparator<in Char>): Char? { if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) { val e = this[i]
        if (min > e) min = e
    }
    return min }
```

Open Source Used In DNAC Platform Arica 13208
CharSequence.minWith(comparator: Comparator<in Char>): Char? {  if (isEmpty()) return null  var min = this[0]  for (i in 1..lastIndex) {  val e = this[i]  if (comparator.compare(min, e) > 0) min = e  }  return min
}

Returns `true` if the char sequence has no characters.

`Sequence.minWith<caret>`

CharSequence.minWith(Predicate<in Char>): Char? {  if (isEmpty()) return null  var min = this[0]  for (i in 1..lastIndex) {  val e = this[i]  if (predicate(e)) min = e  }  return min
}

Returns `true` if no characters match the given predicate.

`Sequence.minWith<caret>`

CharSequence.none(predicate: (Char) -> Boolean): Boolean {  return isEmpty()
}

`Sequence.none<caret>`

CharSequence.noneWithPredicate(predicate: Predicate<Char>): Boolean {  return isEmpty()
}

`Sequence.filter<caret>`

CharSequence.onEach(action: (Char) -> Unit): S {  return apply { for (element in this) action(element) }
}

CharSequence.reduce(operation: (acc: Char, Char) -> Char): Char {  if (isEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")  var accumulator = this[0]  for (index in 1..lastIndex) {  var accumulator = operation(accumulator, this[index])
  }  return accumulator
}

CharSequence.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {  if (isEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
  var accumulator = this[0]  for (index in 1..lastIndex) {  var accumulator = operation(index, accumulator, this[index])
  }  return accumulator
}

CharSequence.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {  if (isEmpty()) return null
  var accumulator = this[0]  for (index in 1..lastIndex) {  var accumulator = operation(accumulator, this[index])
  }  return accumulator
}

CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {  var index = lastIndex  if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
  var accumulator = get(index--) while (index >= 0) {  var accumulator = operation(index, accumulator, get(index--))
  --index
  }  return accumulator
}

CharSequence.reduceRightIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {  var index = lastIndex
if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
  var accumulator = get(index--) while (index >= 0) {  var accumulator = operation(index, get(index--), accumulator)
  --index
  }  return accumulator
}

CharSequence.reduceRightOrNull(operation: (Char, acc: Char) -> Char): Char? {  var index = lastIndex
  if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")
  var accumulator = get(index--) while (index >= 0) {  var accumulator = operation(get(index--), accumulator)
  --index
  }  return accumulator
}

Returns a list containing successive
accumulation values generated by applying [operation] from left to right to each character and current accumulator value that starts with [initial] value. * Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. * * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value. 

@sample samples.collections.Collections.Aggregates.scan

```kotlin
@SinceKotlin("1.3.0")
@ExperimentalStdlibApi
public inline fun <R> CharSequence.scan(initial: R,
    operation: (acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
```

**Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with [initial] value. * * @param [operation] function that takes the index of a character, current accumulator value and the character itself, and calculates the next accumulator value. * * @sample samples.collections.Collections.Aggregates.scanIndexed

```kotlin
@SinceKotlin("1.3.0")
@ExperimentalStdlibApi
public inline fun <R> CharSequence.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    var accumulator = initial
    val result = ArrayList<R>(length + 1).apply { add(initial) }
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence. * * Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. * * @param [operation] function that takes current accumulator value and a character, and calculates the next accumulator value. * * @sample samples.collections.Collections.Aggregates.scanReduce

```kotlin
@SinceKotlin("1.3.0")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduce(operation: (Char, Char) -> Char): List<Char> {
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply { add(accumulator) }
    for (index in 1 until length) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence. * * Note that `acc` value passed to [operation] function should not be mutated; * otherwise it would affect the previous value in resulting list. * * @param [operation] function that takes the index of a character, current accumulator value and the character itself, and calculates the next accumulator value. * * @sample samples.collections.Collections.Aggregates.scanReduceIndexed

```kotlin
@SinceKotlin("1.3.0")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply { add(accumulator) }
    for (index in 1 until length) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Splits this char sequence into a list of strings each not exceeding the given [size]. * The last string in the resulting list may have less characters than the given [size]. * * @param [size] * * @param [operation] function that splits this char sequence into a list of strings each not exceeding the given [size]. * * @sample samples.text.Strings.chunked

```kotlin
@SinceKotlin("1.2")
public fun
```
Open Source Used In DNAC Platform Arica 13211

```kotlin
CharSequence.chunked(size: Int): List<String> {
    return windowed(size, size, partialWindows =
true)\n\n\n\n} \n\n
/**
 * Splits this char sequence into several char sequences each not exceeding the given [size]\n * and applies the given [transform] function to each.\n * Note that the char sequence passed to the [transform] function is ephemeral and is valid
only inside that function.\n * You should not store it or allow it to escape in some way, unless you made a snapshot
of it.\n * The last char sequence may have less characters than the given [size].\n * @param size the number of elements to take in each char sequence, must be positive and can be greater than
the number of elements in this char sequence.\n * @return list of results of the [transform] applied to each char sequence.\n * The last string in the resulting sequence may have less characters than the given [size].\n * @param size the number of elements to take in each string, must be positive and can be greater than
the number of elements in this char sequence.\n */
@SinceKotlin("1.2")
public fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowed(size, size,
    partialWindows = true, transform = transform)\n\n\n\n\n}\n\n```
way, unless you made a snapshot of it.

* Several last char sequences may have less characters than the given 
  char sequence
  * @param size the number of elements to take in each window
  * @param step the number of elements to move the window forward by on each step, by default 1
  * @param partialWindows controls whether or not to keep partial windows in the end if any
  * @param transform function to transform each window

Returns a sequence of snapshots of the window of the given size sliding along this char sequence with the given step, where each snapshot is a string. Several last strings may have less characters than the given size. Both size and step must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window
  * @param step the number of elements to move the window forward by on each step, by default 1
  * @param partialWindows controls whether or not to keep partial windows in the end if any
  * @param transform function to transform each window

Returns a sequence of results of applying the given transform function to each char sequence representing a view over the window of the given size sliding along this char sequence with the given step. Note that the char sequence passed to the transform function is ephemeral and is valid only inside that function. You should not store it or allow it to escape in some way, unless you made a snapshot of it.

* Several last char sequences may have less characters than the given size. Both size and step must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window
  * @param step the number of elements to move the window forward by on each step, by default 1
  * @param partialWindows controls whether or not to keep partial windows in the end if any
  * @param transform function to transform each window

Returns a list of pairs of each two adjacent characters in this char sequence.

The returned list has length of the shortest char sequence. The returned list has length of the shortest char sequence.

* The returned list has length of the shortest char sequence.

* @sample samples.text.Strings.zip
  * @sample samples.text.Strings.zipWithNext
  * @sample samples.collections.TextCollections.zipWithNext
  * @sample samples.Collections.Transformations.zipWithNext
  * @sample InfixFun.zip

Returns a list containing the results of applying the given [transform] function to each pair of two adjacent characters in this char sequence. The returned list is empty if this char sequence contains less than two characters. @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas

```kotlin
@Sample(StringsKt)
val pairsOfChars = "Hi there!"
val transformedPairs = pairsOfChars.zipWithNext { it, next -> it + next }
transformedPairs.forEach { println(it) }
```

*Returns a list containing the results of applying the given [transform] function to each pair of two adjacent characters in this char sequence.*

- The returned list is empty if this char sequence contains less than two characters.

The returned list is empty if this char sequence contains less than two characters.

- The returned list is empty if this char sequence contains less than two characters.

### Since Kotlin("1.2")

```kotlin
public inline fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {
    val size = length - 1
    if (size < 1) return emptyList()
    val result = ArrayList<R>(size)
    for (index in 0 until size) {
        result.add(transform(this[index], this[index + 1]))
    }
    return result
}
```

### Since Kotlin("1.2")

```kotlin
public fun CharSequence.asIterable(): Iterable<Char> {
    if (this is String && isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}
```

### Since Kotlin("1.2")

```kotlin
public fun CharSequence.asSequence(): Sequence<Char> {
    if (this is String && isEmpty()) return emptySequence()
    return Sequence { this.iterator() }
}
```

```kotlin
package kotlin.text

import kotlin.contracts.contract

/**
 * Returns a sub sequence of this char sequence having leading and trailing characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trim
 */
public inline fun CharSequence.trim(predicate: (Char) -> Boolean): CharSequence {
    var startIndex = 0
    var endIndex = length - 1
    var startFound = false

    while (startIndex <= endIndex) {
        val index = if (!startFound) startIndex else endIndex
        val match = predicate(this[index])

        if (!startFound) {
            if (!match)
                startFound = true
            else
                startIndex += 1
        } else {
            if (!match)
                break
            else
                endIndex -= 1
        }
    }

    return subSequence(startIndex, endIndex + 1)
}

/**
 * Returns a string having leading characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trim
 */
public inline fun String.trim(predicate: (Char) -> Boolean): String = (this as CharSequence).trim(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trim
 */
public inline fun CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)

    return"
}

/**
 * Returns a string having leading characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trimStart
 */
public inline fun String.trimStart(predicate: (Char) -> Boolean): String = (this as CharSequence).trimStart(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trimEnd
 */
public inline fun CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices.reversed())
        if (!predicate(this[index]))
            return subSequence(0, index + 1)

    return"
}

/**
 * Returns a string having trailing characters matching the [predicate] removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trimEnd
 */
public inline fun String.trimEnd(predicate: (Char) -> Boolean): String = (this as CharSequence).trimEnd(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading whitespace removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trim
 */
public fun CharSequence.trim(): CharSequence = trim { it in Char::isWhitespace }

/**
 * Returns a string having leading whitespace removed.
 * @sinceKotlin("1.2")
 * @sample samples.collections.Collections.Transformations.trim
 */
public fun String.trim(): String = (this as CharSequence).trim().toString()

```
removed.

public fun CharSequence.trimStart(): CharSequence = trimStart(Char::isWhitespace)

Returns a string having leading whitespace removed.

public fun String.trimStart(): String = (this as CharSequence).trimStart().toString()

Returns a sub sequence of this char sequence having trailing whitespace removed.

public fun CharSequence.trimEnd(): CharSequence = trimEnd(Char::isWhitespace)

Returns a string having trailing whitespace removed.

public fun String.trimEnd(): String = (this as CharSequence).trimEnd().toString()

public fun String.trim(): String = trimStart() + trimEnd()

Returns a string having leading and trailing whitespace removed.

public fun CharSequence.padStart(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0)
        throw IllegalArgumentException("Desired length $length is less than zero.")

    if (length <= this.length)
        return this

    val sb = StringBuilder(length)
    for (i in 1..(length - this.length))
        sb.append(padChar)
    sb.append(this)
    return sb
}

public fun String.padStart(length: Int, padChar: Char = ' '): String =
    (this as CharSequence).padStart(length, padChar).toString()

public fun CharSequence.padEnd(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0)
        throw IllegalArgumentException("Desired length $length is less than zero.")

    if (length <= this.length)
        return this

    val sb = StringBuilder(length)
    sb.append(this)
    for (i in 1..(length - this.length))
        sb.append(padChar)
    return sb
}

public fun String.padEnd(length: Int, padChar: Char = ' '): String =
    (this as CharSequence).padEnd(length, padChar).toString()

public fun CharSequence.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

public fun CharSequence.isEmpty(): Boolean = length == 0

public fun CharSequence.isNotEmpty(): Boolean = length > 0

// implemented differently in JVM and JS\n//public fun String.isBlank(): Boolean = length() == 0 || all { it.isWhitespace() }

public fun CharSequence.isNotBlank(): Boolean = !isBlank()

public fun CharSequence?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

public fun String.isEmpty(): Boolean = length == 0

public fun String.isNotEmpty(): Boolean = length > 0

// implemented differently in JVM and JS\n//public fun String.isBlank(): Boolean = length() == 0 || all { it.isWhitespace() }

public fun CharSequence.isNotNullOrEmpty(): Boolean = length > 0

public fun CharSequence.isNotBlank(): Boolean = length > 0

public fun String.isEmpty(): Boolean = length == 0

public fun String.isNotEmpty(): Boolean = length > 0

// implemented differently in JVM and JS\n//public fun String.isBlank(): Boolean = length() == 0 || all { it.isWhitespace() }

public fun CharSequence.isNotBlank(): Boolean = !isBlank()
samples.text.Strings.stringIsNullOrBlank
*/
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || this.isBlank()
}

/**
 * Iterator for characters of the given char sequence.
 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private var index = 0

    public override fun nextChar(): Char = get(index++)

    public override fun hasNext(): Boolean = index < length
}

/** Returns the string if it is not ‘null’, or the empty string otherwise.
 */
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""

/**
 * Returns this char sequence if it is not empty
 * or the result of calling [defaultValue] function if the char sequence is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns this char sequence if it is not empty
 * or the result of calling [defaultValue] function if the char sequence is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifBlank(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isBlank()) defaultValue() else this

/**
 * Returns a substring specified by the given [range] of indices.
 */
public val CharSequence.indices: IntRange
    get() = 0..length - 1

/**
 * Returns the index of the last character in the char sequence or -1 if it is empty.
 */
public val CharSequence.lastIndex: Int
    get() = this.length - 1

/**
 * Returns `true` if this CharSequence has Unicode surrogate pair at the specified [index].
 */
public fun CharSequence.hasSurrogatePairAt(index: Int): Boolean {
    return index in 0..length - 2
            && this[index].isHighSurrogate()
            && this[index + 1].isLowSurrogate() && this[index + 1].isSurrogate() && this[index + 1].isSurrogate()
}

/**
 * Returns a substring of chars from a range of this char sequence starting at the
 * [startIndex] and ending right before the [endIndex].
 * @param startIndex the start index (inclusive).
 * @param endIndex the end index (exclusive). If not specified, the length of the char sequence is used.
 */
@kotlin.internal.InlineOnly
public inline fun String.substring(startIndex: Int, endIndex: Int = length): String = subSequence(startIndex, endIndex).toString()

/**
 * Returns a substring after the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(index + 1, length)
}

/**
 * Returns a substring before the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(0, index)
}

/**
 * Returns a substring before the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(0, index)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(index + 1, length)
}

/**
 * Returns a substring before the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(0, index)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 * @param delimiter Char, missingDelimiterValue which defaults to the original string.
 */
@kotlin.internal.InlineOnly
public inline fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1)
        missingDelimiterValue
    else
        substring(index + 1, length)
}
original string.

```kotlin
private fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}

/**
 * Returns a substring before the last occurrence of \[delimiter\].
 * If the string does not contain the delimiter, returns \[missingDelimiterValue\] which defaults to the original string.
 */
public fun String.substringBeforeLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring after the last occurrence of \[delimiter\].
 * If the string does not contain the delimiter, returns \[missingDelimiterValue\] which defaults to the original string.
 */
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

/**
 * Returns a char sequence with content of this char sequence where its part at the given range is replaced with the \[replacement\] char sequence.
 * @param range the index of the first character to be replaced.
 * @param replacement the index of the first character after the replacement to keep in the string.
 */
@UseExperimental(ExperimentalStdlibApi::class)
public fun CharSequence.replaceRange(range: IntRange, replacement: CharSequence): CharSequence {
    return toString().replaceRange(range, replacement)
}

/**
 * Replaces the part of string at the given range with the \[replacement\] string.
 * The end index of the \[range\] is included in the part to be replaced.
 */
@kotlin.internal.InlineOnly
public inline fun String.replaceRange(range: IntRange, replacement: CharSequence): String =
    toString().replaceRange(range, replacement).toString()
```

Original Text:

```kotlin
original string.

```
endIndex).toString()"n"**"n * Returns a char sequence with content of this char sequence where its part at the
given [range] is removed."n"**"n * The end index of the [range] is included in the removed part."n" *\n public fun CharSequence.removeRange(range: IntRange): CharSequence = removeRange(range.start, range.endInclusive +
1)"n"**"n * Removes the part of a string at the given [range]."n"**"n * The end index of the [range] is included in
the removed part."n" *\n@kotlin.internal.InlineOnly* public inline fun String.removeRange(range: IntRange): String =
(from this CharSequence).removeRange(range).toString()"n"**"n * If this char sequence starts with the given
[prefix], returns a new char sequence\n * with the prefix removed. Otherwise, returns a new char sequence with
the same characters."n" *\n public fun CharSequence.removePrefix(prefix: CharSequence): CharSequence {
  if (startsWith(prefix)) {
    return subSequence(prefix.length, length)"n"**"n * return subSequence(0,
length)"n"**"n * If this string starts with the given [prefix], returns a copy of this string\n * with the prefix
removed. Otherwise, returns this string."n" *\n public fun String.removePrefix(prefix: CharSequence): String {
  if (startsWith(prefix)) {
    return substring(prefix.length, length)"n"**"n * return this"n"**"n * If this char sequence ends
with the given [suffix], returns a new char sequence\n * with the suffix removed. Otherwise, returns a new char
sequence with the same characters."n" *\n public fun CharSequence.removeSuffix(suffix: CharSequence): CharSequence {
  if (endsWith(suffix)) {
    return subSequence(0, length - suffix.length)"n"**"n * When this char sequence starts
with the given [prefix] and ends with the given [suffix],\n * returns a new char sequence having both the given [prefix] and [suffix] removed.\n * Otherwise returns a new char sequence with
the same characters."n" *\n public fun CharSequence.removeSurrounding(prefix: CharSequence, suffix: CharSequence):
CharSequence {
  if ((length >= prefix.length + suffix.length) && startsWith(prefix) &&
endsWith(suffix)) {
    return subSequence(prefix.length, length - suffix.length)"n"**"n * Removes from a string both the
given [prefix] and [suffix] if and only if\n * it starts with the [prefix] and ends with
the [suffix].\n * Otherwise returns this unchanged."n" *\n public fun String.removeSurrounding(prefix: CharSequence, suffix: CharSequence):
CharSequence {
  if ((length >= prefix.length + suffix.length) &&
startsWith(prefix) &&
endsWith(suffix)) {
    return substring(prefix.length, length - suffix.length)"n"**"n * When this char sequence starts
with and ends with the given [delimiter],\n * returns a new char sequence having this [delimiter] removed both from the start and end.\n * Otherwise returns a new char
sequence with the same characters."n" *\n public fun CharSequence.removeSurrounding(delimiter: CharSequence): CharSequence =
removeSurrounding(delimiter, delimiter)"n"**"n * Removes the given [delimiter] string from both
the start and the end of this string\n * if and only if it starts with and ends with the [delimiter].\n * Otherwise returns this string unchanged."n" *\n public fun String.removeSurrounding(delimiter: CharSequence): String =
removeSurrounding(delimiter, delimiter)"n"**"n * Replace part of string before the first occurrence of given
delimiter with the [replacement] string."n" * If the string does not contain the delimiter, returns
[missingDelimiterValue] which defaults to the original string."n" *\n public fun String.replaceBefore(delimiter: Char, replacement: String, missingDelimiterValue: String =
this): String {
  val index = indexOf(delimiter)"n"**"n * return if (index == -1) missingDelimiterValue else replaceRange(0,
index, replacement)"n"**"n * Replace part of string before the first occurrence of given delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to
the original string."n" *\n public fun String.replaceBefore(delimiter: String, replacement: String, missingDelimiterValue: String =
this): String {
  val index = indexOf(delimiter)"n"**"n * return if (index == -1) missingDelimiterValue else replaceRange(0,
index, replacement)"n"**"n * Replace part of string after the first occurrence of given delimiter with the [replacement]
string.\n * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to
the original string."n" *\n public fun String.replaceAfter(delimiter: Char, replacement: String, missingDelimiterValue: String =
this): String {
  val index = indexOf(delimiter)"n"**"n * return if (index == -1) missingDelimiterValue else replaceRange(index + 1,
length, replacement)"n"**"n * Replace part of string after the first occurrence of given
delimiter with the [replacement] string.\n * If the string does not contain the delimiter, returns

[missingDelimiterValue] which defaults to the original string.

```kotlin
public fun String.replaceAfter(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}
```

Replace part of string after the last occurrence of given delimiter with the [replacement] string.

* If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.

```kotlin
public fun String.replaceAfterLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}
```

Replace part of string before the last occurrence of given delimiter with the [replacement] string.

* If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.

```kotlin
public fun String.replaceBeforeLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}
```

Replace part of string before the last occurrence of given delimiter with the [replacement] string.

* If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.

```kotlin
public fun String.replaceBeforeLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}
```

public fun CharSequence.startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && is String && prefix is String)
        return this.startsWith(prefix)
    else
        return regionMatchesImpl(0, prefix, 0, prefix.length, ignoreCase) // Returns `true` if a substring of this char sequence starting at the specified offset [startIndex] starts with the specified prefix.
}

public fun CharSequence.startsWith(prefix: CharSequence, startIndex: Int, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && prefix is String)
        return this.startsWith(prefix, startIndex)
    else
        return regionMatchesImpl(startIndex, prefix, 0, prefix.length, ignoreCase) // Returns `true` if this char sequence ends with the specified suffix.
}

public fun CharSequence.endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean {
    if (!ignoreCase && this is String && suffix is String)
        return this.endsWith(suffix)
    else
        return regionMatchesImpl(length - suffix.length, suffix, 0, suffix.length, ignoreCase) // common prefix and suffix
}

public fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val shortestLength = minOf(this.length, other.length)
    var i = 0
    while (i < shortestLength && this[i].equals(other[i], ignoreCase)) {
        i++
    }
    if (this.hasSurrogatePairAt(i - 1) || other.hasSurrogatePairAt(i - 1)) {
        i--
    }
    return subSequence(0, i).toString()
}

public fun CharSequence.commonSuffixWith(other: CharSequence, ignoreCase: Boolean = false): String {
    val thisLength = this.length
    val otherLength = other.length
    val shortestLength = minOf(thisLength, otherLength)
    var i = 0
    while (i < shortestLength && this[thisLength - i - 1].equals(other[otherLength - i - 1], ignoreCase)) {
        i++
    }
    if (this.hasSurrogatePairAt(thisLength - i - 1) || other.hasSurrogatePairAt(otherLength - i - 1)) {
        i--
    }
    return subSequence(thisLength - i, thisLength).toString()
}

public fun CharSequence.indexOfAny(chars: CharArray, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    if (!ignoreCase && chars.size == 1 && this is String) {
        val char = chars.single()
        return nativeIndexOf(char, startIndex)
    }
    for (index in startIndex.coerceAtLeast(0)..lastIndex) {
        val charAtIndex = get(index)
        if (chars.any { it.equals(charAtIndex, ignoreCase) })
            return index
    }
    return -1
}

public fun CharSequence.lastIndexOfAny(chars: CharArray, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    if (!ignoreCase && chars.size == 1 && this is String) {
        val char = chars.single()
        return nativeLastIndexOf(char, startIndex)
    }
    for (index in startIndex.coerceAtMost(lastIndex) downTo 0) {
        val charAtIndex = get(index)
        if (chars.any { it.equals(charAtIndex, ignoreCase) })
            return lastIndexOf(char, startIndex)
    }
    return -1
}

private fun CharSequence.indexOf(other: CharSequence, startIndex: Int, endIndex: Int, ignoreCase: Boolean, last: Boolean = false): Int {
    val indices = if (!last)
        startIndex.coerceAtLeast(0)..endIndex.coerceAtMost(length)
    else
        startIndex.coerceAtMost(lastIndex) downTo endIndex.coerceAtLeast(0)

    if (this is String && other is String) { // smart cast
        for (index in indices) {
            if (other.regionMatches(0, this, index, other.length, ignoreCase))
                return index
        }
    }
    return -1
}
if (nextState == -1) {calcNext(); return nextState == 1; } return nextState == 1;

/**
 * Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 * @param delimiters One or more characters to be used as delimiters.
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param startDelimiters The index to start searching delimiters from.
 * @return Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 */
private fun CharSequence.rangesDelimitedBy(delimiters: CharArray, startIndex: Int = 0, limit: Int = 0): Sequence<IntRange> {
    require(limit >= 0, { "Limit must be non-negative, but was $limit." })
    val delimitersList = delimiters.asList()

    return DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })
}

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more strings to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`. Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param startDelimiters The index to start searching delimiters from.
 * @return Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 */
private fun CharSequence.splitToSequence(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =
    rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more characters to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`. Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param startDelimiters The index to start searching delimiters from.
 * @return Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 */
private fun CharSequence.splitToSequence(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0): Sequence<String> =
    rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more strings to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`. Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param startDelimiters The index to start searching delimiters from.
 * @return Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 */
private fun CharSequence.split(vararg delimiters: String, ignoreCase: Boolean = false, limit: Int = 0): List<String> {
    if (delimiters.size == 1) {
        val delimiter = delimiters[0]
        if (!delimiter.isEmpty()) {
            return split(delimiter, ignoreCase, limit)
        }
    }
    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }
}

/**
 * Splits this char sequence to a list of strings around occurrences of the specified [delimiters].
 * @param delimiters One or more strings to be used as delimiters.
 * @param ignoreCase `true` to ignore character case when matching a delimiter. By default `false`. Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 * @param limit The maximum number of substrings to return. Zero by default means no limit is set.
 * @param startDelimiters The index to start searching delimiters from.
 * @return Returns a sequence of index ranges of substrings in this char sequence around occurrences of the specified [delimiters].
 */
private fun CharSequence.split(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0): List<String> {
    if (delimiters.size == 1) {
        val delimiter = delimiters[0]
        if (!delimiter.isEmpty()) {
            return split(delimiter, ignoreCase, limit)
        }
    }
    return rangesDelimitedBy(delimiters, ignoreCase = ignoreCase, limit = limit).map { substring(it) }
}
character case when matching a delimiter. By default `false`.\n* @param limit The maximum number of substrings to return.\n* @public fun CharSequence.split(vararg delimiters: Char, ignoreCase: Boolean = false, limit: Int = 0): List<String> {\n    if (delimiters.size == 1) {\n        return split(delimiters[0].toString(), ignoreCase, limit)\n    }\n    var currentOffset = 0\n    var nextIndex = indexOf(delimiter, currentOffset, ignoreCase)\n    if (nextIndex == -1 || limit == 1) {\n        return listOf(this.toString())\n    }\n    var isLimited = limit > 0\n    val result = ArrayList<String>(if (isLimited) limit.coerceAtMost(10) else 10)\n    do {\n        result.add(substring(currentOffset, nextIndex))\n        currentOffset = nextIndex + delimiter.length\n    } while (nextIndex != -1)\n    return result\n}\n\n/**\n * Splits this char sequence to a list of lines delimited by any of the following character sequences: CRLF, LF or CR.\n */
public fun CharSequence.lines(): List<String> = lineSequence().toList()
String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}

/**
 * Returns a CharArray containing characters of this string or its substring.
 * @param startTheIndex the beginning (inclusive) of the substring, 0 by default.
 * @param endTheIndex the end (exclusive) of the substring, length of this string by default.
 * @throws IndexOutOfBoundsException if [startTheIndex] is less than zero or [endTheIndex] is greater than the length of this string.
 * @throws IllegalArgumentException if [startTheIndex] is greater than [endTheIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.toCharArray(startTheIndex: Int = 0, endTheIndex: Int = this.length): CharArray {
    AbstractList.checkBoundsIndexes(startTheIndex, endTheIndex, length) return CharArray(endTheIndex - startTheIndex) { get(startTheIndex + it) }
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array.
 * Malformed byte sequences are replaced by the replacement char \uFFFD.
 * @param startTheIndex the beginning (inclusive) of the subrange to decode, 0 by default.
 * @param endTheIndex the end (exclusive) of the subrange to decode, size of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement char \uFFFD.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte sequence and [throwOnInvalidSequence] is true.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun ByteArray.decodeToString(): String {
    return decodeUtf8(this, 0, size, false)
}

/**
 * Decodes a string from the bytes in UTF-8 encoding in this array or its subrange.
 * Malformed byte sequences are replaced by the replacement char \uFFFD.
 * @param startIndex the beginning (inclusive) of the subrange to decode, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to decode, size of this array by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed byte sequence or replace it by the replacement char \uFFFD.
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length of this string.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 * @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(): ByteArray {
    return encodeUtf8(this, 0, length, false)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding.
 * Any malformed char sequence is replaced by the replacement byte sequence.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(startIndex: Int = 0, endIndex: Int = this.length, throwOnInvalidSequence: Boolean = false): ByteArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size) return encodeUtf8(this, startIndex, endIndex, throwOnInvalidSequence)
}

/**
 * Returns a copy of this string converted to upper case using the rules of the default locale.
 * @sample samples.text.Strings.toUpperCase
 */
public actual inline fun String.toUpperCase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to lower case using the rules of the default locale.
 * @sample samples.text.Strings.toLowerCase
 */
public actual inline fun String.toLowerCase(): String = asDynamic().toLowerCase()
actual inline fun String.substring(startIndex: Int): String = asDynamic().substring(startIndex)

actual inline fun String.substring(startIndex: Int, endIndex: Int): String = asDynamic().substring(startIndex, endIndex)

actual inline fun String.concat(str: String): String = asDynamic().concat(str)

actual inline fun String.match(regex: String): Array<String>? = asDynamic().match(regex)

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

internal inline fun String.nativeReplace(pattern: RegExp, replacement: String): String = asDynamic().replace(pattern, replacement)

@SinceKotlin("1.2")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {
    if (ignoreCase) {
        val n1 = this.length
        val n2 = other.length
        val min = minOf(n1, n2)
        var start = 0
        while (true) {
            val end = minOf(start + 16, min)
            var s1 = this.substring(start, end)
            var s2 = other.substring(start, end)
            if (s1 != s2) {
                s1 = s1.toUpperCase()
                s2 = s2.toUpperCase()
                if (s1 != s2) {
                    s1 = s1.toLowerCase()
                    s2 = s2.toLowerCase()
                    if (s1 != s2) {
                        return s1.compareTo(s2)
                    }
                }
            }
            if (end == min) break
            start = end
        }
        return n1 - n2
    } else {
        return compareTo(other)
    }
}

private val STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b -> a.compareTo(b, ignoreCase = true) }

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String>
    get() = STRING_CASE_INSENSITIVE_ORDER

// Auto-generated file. DO NOT EDIT!

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
internal class UByteArray
    @Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
    @PublishedApi
    internal constructor(byteArray: ByteArray) : Collection<UByte> {

        /** Creates a new array of the specified [size], with all elements initialized to zero. */
        constructor(size: Int) : this(ByteArray(size))

        /**
         * Returns the array element at the given [index]. This method can be called using the index operator. 
         * * If the [index] is out of bounds of this array, throws an 
         * [IndexOutOfBoundsException] except in Kotlin/JS
         * * where the behavior is unspecified.
         * * \*/
         *
         * Sets the element at the given [index] to the given [value]. This method can be called using the index operator. 
         * * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
         * * where the behavior is unspecified.
         * * \*/
         *
         * Returns the number of elements in the array. */
        override val size: Int get() = byteArray.size

        /** Returns the number of elements in the array. */
        override operator fun iterator(): UByteIterator = Iterator(byteArray)

        private class Iterator(private val byteArray: ByteArray) : UByteIterator() {
            private var index = 0
            override fun hasNext() = index < byteArray.size
            override fun nextUByte() = if (index < byteArray.size) byteArray[index++].toUByte() else throw NoSuchElementException(index.toString())
        }

        override fun contains(element: UByte): Boolean {
            // TODO: Eliminate this check after KT-30016 gets fixed.
            // Currently JS BE does not generate special bridge method for this method.
            if ((element as Any?) !is UByte) return false
            return byteArray.contains(element) 
        }

        override fun containsAll(elements: Collection<UByte>): Boolean { return (elements as Collection<*>).all { it is UByte && byteArray.contains(it) } }

        fun override fun contains(element: UByte): Boolean {
            return (elements as Collection<*>).all { it is UByte && byteArray.contains(it) } 
        }

        // Create a new array of the specified [size], where each element is created by calling the specified [init] function. 
        // The function [init] is called for each array element sequentially starting from the first one.
        fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
            return UByteArray(size, init)
        }

    }

    /** Creates an iterator over the elements of the array. */
    override operator fun iterator(): UByteIterator = Iterator(byteArray)

    fun override fun iterator(): UByteIterator = Iterator(byteArray)

    // Create a new array of the specified [size], where each element is created by calling the specified [init] function. 
    // The function [init] is called for each array element sequentially starting from the first one.
    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }

    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }

    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }

    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }

    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }

    fun override fun create(size: Int, init: (Int) -> UByte): UByteArray {
        return UByteArray(size, init)
    }
init(index).toByte()}
})
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes

public inline fun ubyteArrayOf(vararg elements: UByte): UByteArray = elements

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline class UByteArray

private constructor(@PublishedApi internal val storage: IntArray): Collection<UInt> {

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     */
    public operator fun get(index: Int): UInt = storage[index].toUInt()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
     */
    public operator fun set(index: Int, value: UInt) {
        storage[index] = value.toInt()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(IntArray(size))

    private class Iterator(private val array: IntArray) : UIntIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUInt() = if (index < array.size)
            array[index++].toUInt()
        else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UInt): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is UInt)
            return false

        return storage.contains(element.toInt())
    }

    override fun containsAll(elements: Collection<UInt>): Boolean {
        return (elements as Collection<*>).all { it is UInt && storage.contains(it.toInt()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified [init]
 * function.
 * The function [init] is called for each array element sequentially starting from the first one.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
    return UIntArray(IntArray(size) { index ->
        init(index).toInt()
    })
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes

public inline class ULongArray

private constructor(@PublishedApi internal val storage: LongArray): Collection<ULong> {

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     */
    public operator fun get(index: Int): ULong = storage[index].toULong()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
     */
    public operator fun set(index: Int, value: ULong) {
        storage[index] = value.toLong()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(LongArray(size))

    private class Iterator(private val array: LongArray) : ULongIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextULong() = if (index < array.size)
            array[index++].toULong()
        else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: ULong): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is ULong)
            return false

        return storage.contains(element.toLong())
    }

    override fun containsAll(elements: Collection<ULong>): Boolean {
        return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 *DO NOT EDIT!
 */

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline class ULongArray

private constructor(@PublishedApi internal val storage: LongArray): Collection<ULong> {

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     */
    public operator fun get(index: Int): ULong = storage[index].toULong()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
     */
    public operator fun set(index: Int, value: ULong) {
        storage[index] = value.toLong()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(LongArray(size))

    private class Iterator(private val array: LongArray) : ULongIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextULong() = if (index < array.size)
            array[index++].toULong()
        else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: ULong): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is ULong)
            return false

        return storage.contains(element.toLong())
    }

    override fun containsAll(elements: Collection<ULong>): Boolean {
        return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 *DO NOT EDIT!
 */

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline class ULongArray

private constructor(@PublishedApi internal val storage: LongArray): Collection<ULong> {

    /**
     * Returns the array element at the given [index]. This method can be called using the index operator.
     */
    public operator fun get(index: Int): ULong = storage[index].toULong()

    /**
     * Sets the element at the given [index] to the given [value]. This method can be called using the index operator.
     */
    public operator fun set(index: Int, value: ULong) {
        storage[index] = value.toLong()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates a new array of the specified [size], with all elements initialized to zero. */
    public constructor(size: Int) : this(LongArray(size))

    private class Iterator(private val array: LongArray) : ULongIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextULong() = if (index < array.size)
            array[index++].toULong()
        else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: ULong): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is ULong)
            return false

        return storage.contains(element.toLong())
    }

    override fun containsAll(elements: Collection<ULong>): Boolean {
        return (elements as Collection<*>).all { it is ULong && storage.contains(it.toLong()) }
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}
/** Creates an iterator over the elements of the array. */
public override operator fun iterator(): ULongIterator = 
Iterator(storage)

private class Iterator(private val array: LongArray) : ULongIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextULong() = if (index < array.size) 
        array[index++].toULong() else throw NoSuchElementException(index.toString())
}

override fun contains(element: ULong): Boolean {
    if ((element as Any?) !is ULong)
        return false
    return storage.contains(element.toLong())
}

override fun containsAll(elements: Collection<ULong>): Boolean {
    return (elements as Collection<*>).all { it is ULong && 
        storage.contains(it.toLong()) }
}

override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function. The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(LongArray(size) { index ->
        init(index).toLong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

// Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

// Auto-generated file. DO NOT EDIT!

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class ULong @PublishedApi internal constructor(@PublishedApi internal val data: Long) : Comparable<ULong> {

    companion object {
        /**
         * A constant holding the minimum value an instance of ULong can have.
         */
        public const val MIN_VALUE: ULong = ULong(0)

        /**
         * A constant holding the maximum value an instance of ULong can have.
         */
        public const val MAX_VALUE: ULong = ULong(-1)

        /**
         * The number of bytes used to represent an instance of ULong in a binary form.
         */
        public const val SIZE_BYTES: Int = 8

        /**
         * The number of bits used to represent an instance of ULong in a binary
         * form.
         */
        public const val SIZE_BITS: Int = 64
    }

    /**
     * Compares this value with the specified value for order. Returns zero if this value is equal to
     * the specified other value, a negative number if it's less than other, or a positive number if it's
greater than other. */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order. Returns zero if this value is equal to
     * the specified other value, a negative number if it's less than other, or a positive number if it's
greater than other. */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order. Returns zero if this value is equal to
     * the specified other value, a negative number if it's less than other, or a positive number if it's
greater than other. */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order. Returns zero if this value is equal to
     * the specified other value, a negative number if it's less than other, or a positive number if it's
greater than other. */
    @kotlin.internal.InlineOnly
    public inline operator fun compareTo(other: ULong): Int = 
        ulongCompare(this.data, other.data)

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong = 
        this.plus(other.toULong())
ULong): ULong = ULong(this.data.plus(other.data))
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())
    /** Subtracts the other value from this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = ULong(this.data.minus(other.data))

    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): ULong = this.times(other.toULong())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): ULong = this.times(other.toULong())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): ULong = this.times(other.toULong())
    /** Multiplies this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = ULong(this.data.times(other.data))

    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): ULong = this.div(other.toULong())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): ULong = this.div(other.toULong())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UInt): ULong = this.div(other.toULong())
    /** Divides this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong = ulongDivide(this, other)

    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): ULong = this.rem(other.toULong())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): ULong = this.rem(other.toULong())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): ULong = this.rem(other.toULong())
    /** Calculates the remainder of dividing this value by the other value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = ulongRemainder(this, other)

    /** Increments this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): ULong = ULong(data.inc())
    /** Decrements this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): ULong = ULong(data.dec())

    /** Creates a range from this value to the specified [other] value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: ULong): ULongRange = ULongRange(this, other)

    /** Shifts this value left by the [bitCount] number of bits. */
    @kotlin.internal.InlineOnly
    public inline infix fun shl(bitCount: Int): ULong = ULong(data shl bitCount)
    /** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits with zeros. */
    @kotlin.internal.InlineOnly
    public inline infix fun shr(bitCount: Int): ULong = ULong(data ushr bitCount)

    /** Performs a bitwise AND operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun and(other: ULong): ULong = ULong(this.data and other.data)
    /** Performs a bitwise OR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun or(other: ULong): ULong = ULong(this.data or other.data)
    /** Performs a bitwise XOR operation between the two values. */
    @kotlin.internal.InlineOnly
    public inline infix fun xor(other: ULong): ULong = ULong(this.data xor other.data)
    /** Inverts the bits in this value. */
    @kotlin.internal.InlineOnly
    public inline infix fun not(): ULong = ULong(data.inv())

    /** Converts this [ULong] value to [Byte]. */
    @kotlin.internal.InlineOnly
    public inline fun toByte(): Byte = data.toByte()
    /** Converts this [ULong] value to [Short]. */
    @kotlin.internal.InlineOnly
    public inline fun toShort(): Short = data.toShort()
*An @kotlin.internal.InlineOnly
  public inline fun toShort(): Short = data.toShort() /*@n
  * Converts this [ULong] value to [Int].
  * @n  * If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * The resulting `Int` value is
  * represented by the least significant 32 bits of this `ULong`.
  * @n  * Note that the resulting `Int` value may be negative.
  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toInt(): Int = data.toInt() /*@n
  * Converts this [ULong] value to [Int].
  * @n  * If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int`
  * value represents
  * the same numerical as this `ULong`.
  * Otherwise the result is negative.
  * */@n  * The resulting `Int` value has the same binary representation as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toLong(): Long = data
  * Converts this [ULong] value to [Long].
  * @n  * If this value is less than or equals to [Long.MAX_VALUE], the resulting `Long`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toUByte(): UByte = data.toUByte() /*@n
  * Converts this [ULong] value to [UByte].
  * @n  * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toUShort(): UShort = data.toUShort() /*@n
  * Converts this [ULong] value to [UShort].
  * @n  * If this value is less than or equals to [UShort.MAX_VALUE], the resulting `UShort`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toUInt(): UInt = data.toUInt() /*@n
  * Converts this [ULong] value to [UInt].
  * @n  * If this value is less than or equals to [UInt.MAX_VALUE], the resulting `UInt`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toFloat(): Float = data.toDouble().toFloat() /*@n
  * Converts this [ULong] value to [Float].
  * @n  * If this value is less than or equals to [Float.MAX_VALUE], the resulting `Float`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toDouble(): Double = ulongToDouble(data) /*@n
  * Converts this [ULong] value to [Double].
  * @n  * If this value is less than or equals to [Double.MAX_VALUE], the resulting `Double`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */@n  * @kotlin.internal.InlineOnly
  public inline fun toString(): String = ulongToString(data) /*@n
  * Converts this [ULong] value to [String].
  * @n  * If this value is less than or equals to [String.MAX_VALUE], the resulting `String`
  * value represents
  * the same numerical as this `ULong`.
  * @n  * */
Open Source Used In DNAC Platform Arica 13230

```
[ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.

/*
 * Converts this [Double] value to [ULong].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

/**
 * Converts this [Double] value to [ULong].
 * The fractional part, if any, is rounded down towards zero.
 * Returns zero if this 'Double' value is negative or
 * 'NaN', [ULong.MAX_VALUE] if it's bigger than 'ULong.MAX_VALUE'.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Double.toULong(): ULong = doubleToULong(this)

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found
 * in the license/LICENSE.txt file. */

// Auto-generated file. DO NOT EDIT!

package kotlin

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UShortArray
@PublishedApi
internal constructor(@PublishedApi val storage: ShortArray) : Collection<UShort> {

    /**
     * Returns the array element at the given [index].
     * This method can be called using the index operator.
     * If the [index] is out of bounds of this array, throws an
     * [IndexOutOfBoundsException] except in Kotlin/JS where
     * the behavior is unspecified.
     */
    public operator fun get(index: Int): UShort = storage[index].toUShort()

    /**
     * Sets the element at the given [index] to the given [value].
     * This method can be called using the index operator.
     * If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS
     * where the behavior is unspecified.
     */
    public operator fun set(index: Int, value: UShort) {
        storage[index] = value.toShort()
    }

    /** Returns the number of elements in the array. */
    public override val size: Int get() = storage.size

    /** Creates an iterator over the elements of the array. */
    public override operator fun iterator(): UShortIterator = Iterator(storage)

    private class Iterator(private val array: ShortArray) : UShortIterator() {
        private var index = 0
        override fun hasNext() = index < array.size
        override fun nextUShort() = if (index < array.size) array[index++].toUShort() else throw NoSuchElementException(index.toString())
    }

    override fun contains(element: UShort): Boolean {
        // TODO: Eliminate this check after KT-30016 gets fixed.
        // Currently JS BE does not generate special bridge method for this method.
        if ((element as Any?) !is UShort) return false

        return storage.contains(element.toShort())
    }

    override fun containsAll(elements: Collection<UShort>): Boolean {
        return elements.all { it is UShort &&
        return storage.contains(element.toShort()) } || throw NoSuchElementException(elements.toString())
    }

    override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 * The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UIntArray.component1(): UInt {
    return get(0)
}
```

---

Open Source Used In DNAC Platform Arica 13230
If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS

* @SinceKotlin("1.3")
  @ExperimentalUnsignedTypes
  @kotlin.internal.InlineOnly
  @public inline operator fun ULongArray.component1(): ULong {
return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component1(): UByte {
return get(0)
}

/**
 * Returns 1st *element* from the array.
 * 
 * If the size of this array is less than 1, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component1(): UShort {
return get(0)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component2(): UInt {
return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component2(): ULong {
return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component2(): UByte {
return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * 
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component2(): UShort {
return get(1)
}

/**
 * Returns 3rd *element* from the array.
 * 
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component3(): UInt {
return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * 
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component3(): ULong {
return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * 
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component3(): UByte {
return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * 
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component3(): UShort {
return get(2)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component4(): UInt {
return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component4(): ULong {
return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component4(): UByte {
return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component4(): UShort {
return get(3)
}
size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun ULongArray.component4(): ULong {\n    return get(3)\n}\n This returns 4th element from the array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun UByteArray.component4(): UByte {\n    return get(3)\n}\n This returns 4th element from the array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun UShortArray.component4(): UShort {\n    return get(3)\n}\n This returns 4th element from the array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun UIntArray.component5(): UInt {\n    return get(4)\n}\n This returns 5th element from the array. \n If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@.ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun ULongArray.component5(): ULong {\n    return get(4)\n}\n This returns 5th element from the array. \n If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun UByteArray.component5(): UByte {\n    return get(4)\n}\n This returns 5th element from the array. \n If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) inline operator fun UShortArray.component5(): UShort {\n    return get(4)\n}\n This returns 5th element from the array. \n If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) expect fun UIntArray.elementAt(index: Int): UInt {\n    return if (index >= 0 && index <= lastIndex) get(index) \n else defaultValue(index)\n}\n This returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) expect fun ULongArray.elementAt(index: Int): ULong {\n    return if (index >= 0 && index <= lastIndex) get(index) \n else defaultValue(index)\n}\n This returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) expect fun UByteArray.elementAt(index: Int): UByte {\n    return if (index >= 0 && index <= lastIndex) get(index) \n else defaultValue(index)\n}\n This returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\(public\) expect fun UShortArray.elementAt(index: Int): UShort {\n    return if (index >= 0 && index <= lastIndex) get(index) \n else defaultValue(index)\n}\n This returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array. \n If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS \n where the behavior is unspecified.
index <= lastIndex) get(index) else defaultValue(index) * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

/*
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 * 
 * @sample
 * samples.collections.Collections.Elements.elementAtOrElse
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

/*
 * Returns an element at the given [index] or null if the [index] is out of bounds of this array.
 * 
 * @sample
 * samples.collections.Collections.Elements.elementAtOrNull
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.elementAtOrNull(index: Int): UInt? {
    return this.getOrNull(index)
}

/*
 * Returns the first element matching the given [predicate], or null if no such element was found.
 * 
 * @sample
 * samples.collections.Collections.Elements.find
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.find(predicate: (UInt) -> Boolean): UInt? {
    return firstOrNull(predicate)
}

/*
 * Returns the last element matching the given [predicate], or null if no such element was found.
 * 
 * @sample
 * samples.collections.Collections.Elements.findLast
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.findLast(predicate: (UInt) -> Boolean): UInt? {
    return lastOrNull(predicate)
}
Returns the last element matching the given [predicate], or `null` if no such element was found.

```kotlin
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UShortArray.findLast(predicate: (UShort) -> Boolean): UShort? { `n   return lastOrNull(predicate)\`}n\n**\n* Returns first element.\n* @throws [NoSuchElementException] if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UIntArray.firstOrNull(): UInt? { `n   return storage.first().toUInt()\`}n\n**\n* Returns first element.\n* @throws [NoSuchElementException] if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun ULongArray.first(): ULong { `n   return storage.first().toULong()\`}n\n**\n* Returns first element.\n* @throws [NoSuchElementException] if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UByteArray.first(): UByte { `n   return storage.first().toUByte()\`}n\n**\n* Returns first element.\n* @throws [NoSuchElementException] if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UShortArray.first(): UShort { `n   return storage.first().toUShort()\`}n\n**\n* Returns first element.\n* @throws [NoSuchElementException] if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UIntArray.firstOrNull(predicate: (UInt) -> Boolean): UInt? { `n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException("Array contains no element matching the predicate.\ transformative\")\`}n\n**\n* Returns the first element matching the given [predicate].\n* @throws [NoSuchElementException] if no such element is found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? { `n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException("Array contains no element matching the predicate.\ transformative\")\`}n\n**\n* Returns the first element matching the given [predicate].\n* @throws [NoSuchElementException] if no such element is found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? { `n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException("Array contains no element matching the predicate.\ transformative\")\`}n\n**\n* Returns the first element matching the given [predicate].\n* @throws [NoSuchElementException] if no such element is found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? { `n   for (element in this) if (predicate(element)) return element\n   throw NoSuchElementException("Array contains no element matching the predicate.\ transformative\")\`}n\n**\n* Returns the first element, or `null` if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun UIntArray.firstOrNull(): UInt? { `n   if (isEmpty()) null else this[0]\`}n\n**\n* Returns the first element, or `null` if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun ULongArray.firstOrNull(): ULong? { `n   if (isEmpty()) null else this[0]\`}n\n**\n* Returns the first element, or `null` if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun UByteArray.firstOrNull(): UByte? { `n   if (isEmpty()) null else this[0]\`}n\n**\n* Returns the first element, or `null` if the array is empty.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun UShortArray.firstOrNull(): UShort? { `n   if (isEmpty()) null else this[0]\`}n\n**\n* Returns the first element, or `null` if the array was not found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun UIntArray.firstOrNull(predicate: (UInt) -> Boolean): UInt? { `n   for (element in this) if (predicate(element)) return element\n   return null\`}n\n**\n* Returns the first element matching the given [predicate], or `null` if element was not found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? { `n   for (element in this) if (predicate(element)) return element\n   return null\`}n\n**\n* Returns the first element matching the given [predicate], or `null` if element was not found.\n
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? { `n   for (element in this) if (predicate(element)) return element\n   return null\`}n\n**\n* Returns the first element matching the given [predicate], or `null` if element was not found.
inline fun ULongArray.firstOrNull(predicate: (ULong) -> Boolean): ULong? {
    for (element in this) if (predicate(element)) return element
    return null
} * Returns the first element matching the given [predicate], or `null` if element was not found.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.firstOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (element in this) if (predicate(element)) return element
    return null
} * Returns the first element matching the given [predicate], or `null` if element was not found.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.firstOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (element in this) if (predicate(element)) return element
    return null
} * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.getOrElse(index: Int, defaultValue: (Int) -> UInt): UInt {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.getOrElse(index: Int, defaultValue: (Int) -> ULong): ULong {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.getOrNull(index: Int): UInt? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.getOrNull(index: Int): ULong? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.getOrNull(index: Int): UByte? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.getOrNull(index: Int): UShort? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
} * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.indexOf(element: UInt): Int {
    return storage.indexOf(element.toInt())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOf(element: UByte): Int {
    return storage.indexOf(element.toByte())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOf(element: UShort): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.indexOf(element: UInt): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.indexOf(element: ULong): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOf(element: UByte): Int {
    return storage.indexOf(element.toByte())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOf(element: UShort): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOf(element: UByte): Int {
    return storage.indexOf(element.toByte())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOf(element: UShort): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOf(element: UByte): Int {
    return storage.indexOf(element.toByte())
} * Returns first index of [element], or -1 if the array does not contain element.

*@
Since Kotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOf(element: UShort): Int {
    return storage.indexOf(element.toLong())
} * Returns first index of [element], or -1 if the array does not contain element.
UShortArray.indexOf(element: UShort): Int {
    var index = storage.indexOf(element.toShort())
    if (index == -1) {
        // code to handle case where element not found
    }
    return index
}

UShortArray.indexOfFirst(predicate: (UShort) -> Boolean): Int {
    var index = -1
    for (i in storage.indices) {
        if (predicate(storage[i].toShort())) {
            index = i
            break
        }
    }
    return index
}

UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {
    var index = -1
    for (i in storage.indices.reversed()) {
        if (predicate(storage[i].toShort())) {
            index = i
            break
        }
    }
    return index
}

UByteArray.indexOf(element: UByte): Int {
    var index = storage.indexOf(element.toUByte())
    if (index == -1) {
        // code to handle case where element not found
    }
    return index
}

UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {
    var index = -1
    for (i in storage.indices) {
        if (predicate(storage[i].toUByte())) {
            index = i
            break
        }
    }
    return index
}

UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {
    var index = -1
    for (i in storage.indices.reversed()) {
        if (predicate(storage[i].toUByte())) {
            index = i
            break
        }
    }
    return index
}

UByteArray.indexOf(element: UByte): Int {
    var index = storage.indexOf(element.toUByte())
    if (index == -1) {
        // code to handle case where element not found
    }
    return index
}

UByteArray.indexOfFirst(predicate: (UByte) -> Boolean): Int {
    var index = -1
    for (i in storage.indices) {
        if (predicate(storage[i].toUByte())) {
            index = i
            break
        }
    }
    return index
}

UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {
    var index = -1
    for (i in storage.indices.reversed()) {
        if (predicate(storage[i].toUByte())) {
            index = i
            break
        }
    }
    return index
}

UIntArray.indexOf(element: UInt): Int {
    var index = storage.indexOf(element.toUInt())
    if (index == -1) {
        // code to handle case where element not found
    }
    return index
}

UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int {
    var index = -1
    for (i in storage.indices) {
        if (predicate(storage[i].toUInt())) {
            index = i
            break
        }
    }
    return index
}

UIntArray.indexOfLast(predicate: (UInt) -> Boolean): Int {
    var index = -1
    for (i in storage.indices.reversed()) {
        if (predicate(storage[i].toUInt())) {
            index = i
            break
        }
    }
    return index
}

ULongArray.indexOf(element: ULong): Int {
    var index = storage.indexOf(element.toULong())
    if (index == -1) {
        // code to handle case where element not found
    }
    return index
}

ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int {
    var index = -1
    for (i in storage.indices) {
        if (predicate(storage[i].toULong())) {
            index = i
            break
        }
    }
    return index
}

ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {
    var index = -1
    for (i in storage.indices.reversed()) {
        if (predicate(storage[i].toULong())) {
            index = i
            break
        }
    }
    return index
}
public inline fun ULongArray.last(predicate: (ULong) -> Boolean): ULong {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun UByteArray.last(predicate: (UByte) -> Boolean): UByte {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last element matching the given [predicate].
 * @throws NoSuchElementException if no such element is found.
 */
public inline fun UShortArray.last(predicate: (UShort) -> Boolean): UShort {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public inline fun UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf(element.toInt())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public inline fun ULongArray.lastIndexOf(element: ULong): Int {
    return storage.lastIndexOf(element.toLong())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public inline fun UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf(element.toByte())
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun UIntArray.lastOrNull(): UInt? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun ULongArray.lastOrNull(): ULong? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun UIntArray.lastOrNull(predicate: (UInt) -> Boolean): UInt? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun ULongArray.lastOrNull(predicate: (ULong) -> Boolean): ULong? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun UByteArray.lastOrNull(predicate: (UByte) -> Boolean): UByte? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.lastOrNull(predicate: (UShort) -> Boolean): UShort? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.random(): ULong {
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.random(): UByte {
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.random(): UShort {
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.randomOrNull(): UByte? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this array using the specified source of randomness.
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.randomOrNull(): UShort? {
    return randomOrNull(Random)
}
ULongArray.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns a random element from this array using the specified source of
 * randomness, or `null` if this array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public fun ULongArray.randomOrNull(random: Random): UByte? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

/**
 * Returns the single element, or throws an exception if the array is empty or
 * has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@public inline fun UByteArray.single(): UByte {
    if (isEmpty())
        throw NoSuchElementException("Array contains no element matching the predicate.")
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
}
(found) throw NoSuchElementException("Array contains no element matching the predicate.")
@Suppress("UNCHECKED_CAST")
    return single as UShort
}

/**
 * Returns single element, or `null` if
 * the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.singleOrNull(): UInt? {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single
 * element matching the given [predicate], or `null` if element was not found or more than one element was found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.drop(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.singleOrNull(): ULong? {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.drop(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.drop(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/**
 * Returns a list containing all elements except first [n] elements.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.drop(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))
}

/*@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.singleOrNull(): UInt? {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}*/

/*@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.singleOrNull(): ULong? {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}*/

Open Source Used In DNAC Platform Arica  13240
samples.collections.Collections.Transformations.drop
*\n@SinceKotlin("1.3.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.drop(n: Int): List<UShort> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return takeLast((size - n).coerceAtLeast(0))\n}\n/**\n * Returns a list containing all elements except last [n] elements.
 * \n * @throws IllegalArgumentException if [n] is negative.\n * @sample samples.collections.Collections.Transformations.drop
 */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.dropLast(n: Int): List<UInt> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n}\n/**\n * Returns a list containing all elements except last [n] elements.
 * \n * @throws IllegalArgumentException if [n] is negative.\n * @sample samples.collections.Collections.Transformations.drop
 */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.dropLast(n: Int): List<ULong> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n}\n/**\n * Returns a list containing all elements except last [n] elements that satisfy the given [predicate].
 * \n * @sample samples.collections.Collections.Transformations.drop
 */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.dropLast(n: Int): List<UByte> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n}\n/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * \n * @sample samples.collections.Collections.Transformations.drop
 */\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UShortArray.dropLast(n: Int): List<UShort> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n}
public inline fun ULongArray.dropWhile(predicate: (ULong) -> Boolean): List<ULong> {
    var yielding = false
    val list = ArrayList<ULong>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), predicate)
}

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.filterIndexed(predicate: (Int, ULong) -> Boolean): List<ULong> {
    return filterIndexedTo(ArrayList<ULong>(), predicate)
}
**Open Source Used in DNAC Platform Arica**

```kotlin
UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> { return filterIndexedTo(ArrayList<UByte>(), predicate) }

/**
 * Returns a list containing only elements matching the given [predicate].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> { return filterIndexedTo(ArrayList<UShort>(), predicate) }

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {
    return filterNotTo(destination, predicate) }  // Append all elements not matching the given [predicate] to the given [destination].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
    return filterNotTo(destination, predicate) }  // Append all elements not matching the given [predicate] to the given [destination].
```

*Sample data*

```
public class DNACPlatformArina {
    public static void main(String[] args) {
        UByteArray byteArray = UByteArray.of(UByte.values());  // Create a byte array
        UByteArray filteredArray = byteArray.filterIndexed { it == 1 }  // Filter by value
        println(filteredArray)  // Output filtered array
    }
}
```
public inline fun <C : 
MutableCollection<in UInt>> UIntArray.filterNotTo(destination: C, predicate: (UInt) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in ULong>> ULongArray.filterNotTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in UByte>> UByteArray.filterNotTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements not matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in UShort>> UShortArray.filterNotTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (!predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (UInt) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (ULong) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in UByte>> UByteArray.filterTo(destination: C, predicate: (UByte) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all
 * elements matching the given [predicate] to the given [destination].
 */
public inline fun <C : 
MutableCollection<in UShort>> UShortArray.filterTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list
 * containing elements at indices in the specified [indices] range.
 */
public fun UIntArray.slice(indices: IntRange): List<UInt> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
public fun ULongArray.slice(indices: IntRange): List<ULong> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun UByteArray.slice(indices: Iterable<Int>): List<UByte> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UByte>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
public fun UShortArray.slice(indices: Iterable<Int>): List<UShort> {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    val list = ArrayList<UShort>(size)
    for (index in indices) {
        list.add((get(index))
    }
    return list
}
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun ULongArray.slice(indices: Iterable\<Int\>): List\<ULong\> {
    var size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    var list = ArrayList\<ULong\>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.slice(indices: Iterable\<Int\>): List\<UByte\> {
    var size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    var list = ArrayList\<UByte\>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns a list containing elements at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.slice(indices: Iterable\<Int\>): List\<UShort\> {
    var size = indices.collectionSizeOrDefault(10)
    if (size == 0) return emptyList()
    var list = ArrayList\<UShort\>(size)
    for (index in indices) {
        list.add(get(index))
    }
    return list
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sliceArray(indices: Collection\<Int\>): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: Collection\<Int\>): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: Collection\<Int\>): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements of this array at specified [indices].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: Collection\<Int\>): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.take(n: Int): List\<UInt\> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList\<UInt\>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.take(n: Int): List\<ULong\> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList\<ULong\>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.take(n: Int): List\<UByte\> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList\<UByte\>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.take(n: Int): List\<UShort\> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList\<UShort\>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}
fun UByteArray.take(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<UByte>()
    if (n >= size) return toList()<UByte>()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UByte>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

fun UShortArray.take(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<UShort>()
    if (n >= size) return toList()<UShort>()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UShort>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

fun UIntArray.takeLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<UInt>()
    val size = size
    if (n >= size) return toList()<UInt>()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UInt>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

fun ULongArray.takeLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<ULong>()
    val size = size
    if (n >= size) return toList()<ULong>()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

fun UByteArray.takeLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<UByte>()
    val size = size
    if (n >= size) return toList()<UByte>()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()<UShort>()
    val size = size
    if (n >= size) return toList()<UShort>()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UShort>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

fun UIntArray.takeLastWhile(predicate: (UInt) -> Boolean): List<UInt> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()<UInt>()
}

fun ULongArray.takeLastWhile(predicate: (ULong) -> Boolean): List<ULong> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()<ULong>()
}

fun UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()<UByte>()
}

fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()<UShort>()
}
UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> { for (index in lastIndex downTo 0) |
\n if (!predicate(this[index])) |
\n return drop(index + 1) |
\n return toList().n
\n * @sample samples.collections.Collections.Transformations.take
* \n
UByteArray.reverse(): Unit { storage.reverse().n
\n * Reverses elements in the array in-place.
* \n
UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> { for (index in lastIndex downTo 0) |
\n if (!predicate(this[index])) |
\n return drop(index + 1) |
\n return toList().n
\n * Returns a list containing first elements satisfying the given [predicate].n * @sample samples.collections.Collections.Transformations.take
* \n
ULongArray.reverse(): Unit { storage.reverse().n
\n * Reverses elements in the array in-place.
* \n
UIntArray.reverse(): Unit { storage.reverse().n
\n * Reverses elements in the array in-place.
* \n
UIntArray.reversed(): List<UInt> { if (isEmpty()) return emptyList().n
\n * Returns a list with elements in reversed order.
* \n
UIntArray.reversedArray(): UIntArray { \n \n if (isEmpty()) return emptyList().n
\n * Returns a list with elements of this array in reversed order.
* \n
UByteArray.reverse(): Unit { storage.reverse().n
\n * Reverses elements in the array in-place.
* \n
UByteArray.reversed(): List<UByte> { if (isEmpty()) return emptyList().n
\n * Returns a list with elements in reversed order.
* \n
UByteArray.reversedArray(): UByteArray { return UlntArray(storage.reversedArray()).n
\n * Returns an array with elements of this array in reversed order.
* \n
UByteArray: UByteArray { return UlntArray(storage.reversedArray()).n
\n * Returns an
array with elements of this array in reversed order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun ULongArray.reversedArray(): ULongArray {
    return ULongArray(storage.reversedArray())
}
```

Returns an array with elements of this array in reversed order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UByteArray.reversedArray(): UByteArray {
    return UByteArray(storage.reversedArray())
}
```

Returns an array with elements of this array in reversed order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public inline fun UShortArray.reversedArray(): UShortArray {
    return UShortArray(storage.reversedArray())
}
```

Returns an array with elements of this array in reversed order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UIntArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

Sorts elements in the array in-place descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun ULongArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

Sorts elements in the array in-place descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UByteArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

Sorts elements in the array in-place descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UShortArray.sortDescending(): Unit {
    if (size > 1) {
        sort()
        reverse()
    }
}
```

Sorts elements in the array in-place descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UIntArray.sorted(): List<UInt> {
    return copyOf().apply { sort() }.asList()
}
```

Returns a list of all elements sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun ULongArray.sorted(): List<ULong> {
    return copyOf().apply { sort() }.asList()
}
```

Returns a list of all elements sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UByteArray.sorted(): List<UByte> {
    return copyOf().apply { sort() }.asList()
}
```

Returns a list of all elements sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UShortArray.sorted(): List<UShort> {
    return copyOf().apply { sort() }.asList()
}
```

Returns a list of all elements sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UIntArray.sortedArray(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

Returns an array with all elements of this array sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun ULongArray.sortedArray(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

Returns an array with all elements of this array sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UByteArray.sortedArray(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

Returns an array with all elements of this array sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UShortArray.sortedArray(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sort() }
}
```

Returns an array with all elements of this array sorted according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UIntArray.sortedArrayDescending(): UIntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

Returns an array with all elements of this array sorted descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun ULongArray.sortedArrayDescending(): ULongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

Returns an array with all elements of this array sorted descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UByteArray.sortedArrayDescending(): UByteArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

Returns an array with all elements of this array sorted descending according to their natural sort order.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@public fun UShortArray.sortedArrayDescending(): UShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}
```

Returns an array with all elements of this array sorted descending according to their natural sort order.
Returns a list of all elements sorted descending according to their natural sort order.

* The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

* Returns a list of all elements sorted descending according to their natural sort order.

* The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.

* Returns an array of type [ByteArray], which is a view of this array where each element is a signed reinterpretation of the corresponding element of this array.

* Returns a list of type [List] that wraps the original array.

* Returns a list of type [List] that wraps the original array.

* Returns an array of type [LongArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

* Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

* Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

* Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

* Returns a list of type [List] that wraps the original array.

* Returns a list of type [List] that wraps the original array.

* Returns a list of type [List] that wraps the original array.

* Returns a list of type [List] that wraps the original array.

* Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.

* Returns an array of type [UByteArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.
public inline fun LongArray.asULongArray(): ULongArray {
    return ULongArray(this)
}

/**
 * Returns an array of type [ULongArray], which is a view of this array where each element is an unsigned reinterpretation of the corresponding element of this array.
 */
public inline fun ShortArray.asUShortArray(): UShortArray {
    return UShortArray(this)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 */
public infix fun UIntArray.contentEquals(other: UIntArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 */
public infix fun ULongArray.contentEquals(other: ULongArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 */
public infix fun UByteArray.contentEquals(other: UByteArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns `true` if the two specified arrays are *structurally* equal to one another, i.e. contain the same number of the same elements in the same order.
 */
public infix fun UShortArray.contentEquals(other: UShortArray): Boolean {
    return storage.contentEquals(other.storage)
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
public fun UIntArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
public fun ULongArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
public fun UByteArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is [List].
 */
public fun UShortArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
public fun UIntArray.copyOf(destination: UIntArray, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    // Implementation
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
public fun ULongArray.copyOf(destination: ULongArray, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    // Implementation
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
public fun UByteArray.copyOf(destination: UByteArray, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    // Implementation
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 */
public fun UShortArray.copyOf(destination: UShortArray, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    // Implementation
}
starting at the specified [destinationOffset].
* or when that index is out of the [destination] array indices range.
* 
  * @return the [destination] array.

*\n*\n@SinceKotlin("1.3\")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UIntArray.copyInto(destination: UIntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UIntArray
{
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

*\n*\n**Copies this array or its subrange into the [destination] array and returns that array.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
*\n*\n* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

**@return the [destination] array.

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun ULongArray.copyInto(destination: ULongArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ULongArray
{
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

*\n*\n**Copies this array or its subrange into the [destination] array and returns that array.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
*\n*\n* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

**@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].

**@return the [destination] array.

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UByteArray
{
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

*\n*\n**Copies this array or its subrange into the [destination] array and returns that array.

It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
*\n*\n* @param destination the array to copy to.
* @param destinationOffset the position in the [destination] array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
* @param endIndex (exclusive) of the subrange to copy, size of this array by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.

**@throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].

**@return the [destination] array.

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UShortArray
{
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.copyOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UIntArray.copyOf(): UIntArray
{
    return UIntArray(storage.copyOf())
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.copyOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun ULongArray.copyOf(): ULongArray
{
    return ULongArray(storage.copyOf())
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.copyOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UIntArray.compress(): UIntArray
{
    return UIntArray(storage.compress())
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.compressOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun ULongArray.compress(): ULongArray
{
    return ULongArray(storage.compress())
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.compressOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun UIntArray.withCapacity(newCapacity: Int): UIntArray
{
    return UIntArray(capacity = newCapacity)
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.withCapacityOf

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n\npublic inline fun ULongArray.withCapacity(newCapacity: Int): ULongArray
{
    return ULongArray(capacity = newCapacity)
}

*\n*\n**Returns new array which is a copy of the original array.

@sample samples.collections.Arrays.CopyOfOperations.withCapacityOf
which is a copy of the original array.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOf(): UByteArray {
    return UByteArray(storage.copyOf())
}
```

Returns new array which is a copy of the original array. The copy is either truncated or padded at the end with zero values if necessary. * @sample samples.collections.Arrays.CopyOfOperations.copyOf

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UIntArray.copyOf(newSize: Int): UIntArray {
    return UIntArray(storage.copyOf(newSize))
}
```

Returns new array which is a copy of the original array, resized to the given [newSize]. The copy is either truncated or padded at the end with zero values if necessary. * @sample samples.collections.Arrays.CopyOfOperations.copyOf

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.copyOfRange(fromIndex: Int, toIndex: Int): UByteArray {
    return UByteArray(storage.copyOfRange(fromIndex, toIndex))
}
```

Returns new array which is a copy of the specified range of the original array. The copy is either truncated or padded at the end of the range (inclusive), must be in `0..array.size`
```kotlin
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun
UShortArray.copyOfRange(fromIndex: Int, toIndex: Int): UShortArray {\n    return UShortArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Fills this array or its subrange with the specified [element] value.\n *\n * @param fromIndex the start of the range (inclusive), 0 by default.\n *\n * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.\n *\n * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toShort(), fromIndex, toIndex)
}

/**
 * Returns the range of valid indices for the array.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the last valid index for the array.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.lastIndex: Int
    get() = storage.lastIndex
```

Open Source Used In DNAC Platform Arica 13253
operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage +
        element.toLong())
} // Returns an array containing all elements of the original array and then the given
[element].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage +
        element.toByte())
} // Returns an array containing all elements of the original array and then the given
[element].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return UShortArray(storage +
        element.toShort())
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return UIntArray(result)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return ULongArray(result)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.plus(elements: UIntArray): UIntArray {
    return UIntArray(storage + elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.plus(elements: ULongArray): ULongArray {
    return ULongArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.plus(elements: UByteArray): UByteArray {
    return UByteArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.plus(elements: UShortArray): UShortArray {
    return UShortArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.plus(elements: UIntArray): UIntArray {
    return UIntArray(storage + elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.plus(elements: ULongArray): ULongArray {
    return ULongArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.plus(elements: UByteArray): UByteArray {
    return UByteArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.plus(elements: UShortArray): UShortArray {
    return UShortArray(storage +
        elements.storage)
} // Returns an array containing all elements of the original array and then all elements
of the given [elements].

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1)
} // Sorts the array in-place.
which is a copy of this array where each element is a signed reinterpretation
* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()\n}

/**
 * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.toIntArray(): IntArray {
    return storage.copyOf()\n}

/**
 * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.toLongArray(): LongArray {
    return storage.copyOf()\n}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UIntArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] }\n}

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun ULongArray.toTypedArray(): Array<ULong> {
    return Array(size) { index -> this[index] }\n}

/**
 * Returns an array of UByte containing all of the elements of this generic array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic fun UByteArray.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }\n}

/**
 * Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned
 * reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())\n}

/**
 * Returns an array of type [UIntArray], which is a copy of this array where each element is an unsigned
 * reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.toUIntArray(): UIntArray {
    return UIntArray(this.copyOf())\n}

/**
 * Returns an array of type [ULongArray], which is a copy of this array where each element is an unsigned
 * reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.toULongArray(): ULongArray {
    return ULongArray(this.copyOf())\n}

/**
 * Returns an array of type [UShortArray], which is a copy of this array where each element is an unsigned
 * reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.toUShortArray(): UShortArray {
    return UShortArray(size) { index -> this[index] }\n}

/**
 * Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned
 * reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()\n}

/**
 * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun IntArray.toIntArray(): IntArray {
    return storage.copyOf()\n}

/**
 * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun LongArray.toLongArray(): LongArray {
    return storage.copyOf()\n}

/**
 * Returns an array of type [ShortArray], which is a copy of this array where each element is a signed reinterpretation\n* of the corresponding element of this array.\n
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ShortArray.toShortArray(): ShortArray {
    return storage.copyOf()\n}
corresponding element of this array.

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun ShortArray.toUShortArray(): UShortArray {
    return UShortArray(this.copyOf())
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R> UIntArray.flatMap(transform: (UInt) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R> ULongArray.flatMap(transform: (ULong) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R> UShortArray.flatMap(transform: (UShort) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R> MutableList<in R>> UIntArray.flatMapTo(destination: C, transform: (UInt) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R, C : MutableCollection<in R>> UByteArray.flatMapTo(destination: C, transform: (UByte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R, C : MutableCollection<in R>> UShortArray.flatMapTo(destination: C, transform: (UShort) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

*\@SinceKotlin("1.3\")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <K> UIntArray.groupBy(keySelector: (UInt) -> K): Map<K, List<UInt>> {
    return groupByTo(LinkedHashMap<K,
MutableList<UInt>()

/**
 * Groups elements of the original array by the key returned by the
 * given [keySelector] function. Returns a map where each group key is associated with
 * a list of corresponding elements. The returned map preserves the entry iteration order of the keys produced
 * from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K> ULongArray.groupBy(keySelector: (ULong) -> K): Map<K, List<ULong>> {
    return groupByTo(LinkedHashMap<K, MutableList<ULong>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map
 * where each group key is associated with a list of corresponding elements. The returned map preserves
 * the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K> UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> {
    return groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and returns a map
 * where each group key is associated with a list of corresponding elements. The returned map preserves
 * the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K> UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given
 * [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values. The returned map preserves
 * the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given
 * [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values. The returned map preserves
 * the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given
 * [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values. The returned map preserves
 * the entry iteration order of the keys produced from the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}
samples.collections.Collections.Transformations.groupByKeysAndValues

*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun <K, V>
UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>>
{\n    \n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, M : MutableMap<in K, MutableList<UInt>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M
{\n    \n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<UInt>()\n        }\n        list.add(element)\n    }\n    return destination\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, M : MutableMap<in K, MutableList<ULong>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K): M
{\n    \n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<ULong>()\n        }\n        list.add(element)\n    }\n    return destination\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, M : MutableMap<in K, MutableList<UByte>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K): M
{\n    \n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<UByte>()\n        }\n        list.add(element)\n    }\n    return destination\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, M : MutableMap<in K, MutableList<UShort>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K): M
{\n    \n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<UShort>()\n        }\n        list.add(element)\n    }\n    return destination\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, V, M : MutableMap<in K, MutableList<V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K, valueTransform: (UInt) -> V): M
{\n    \n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\n\n*\n*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic fun <K, V, M : MutableMap<in K, MutableList<V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K, valueTransform: (ULong) -> V): M
{\n    \n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n\nOpen Source Used In DNAC Platform Arica 13258
destination = Groups values returned by the [valueTransform] function applied to each element of the original array\n* by the key returned by the given [keySelector] function applied to the element\n* and puts to the [destination] map each group key associated with a list of corresponding values.\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline fun <K, V, M : MutableMap<K, MutableList<V>>, UByteArray, UShortArray> groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {
    for (element in this) {
        val key = keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<V>() }\n        list.add(valueTransform(element))\n    }\n    return destination\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.mapIndexed(transform: (Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element and its index in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.mapIndexed(transform: (Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array\n* by the key returned by the given [keySelector] function applied to the element\n* and puts to the [destination] map each group key associated with a list of corresponding values.\n* @return The [destination] map.\n* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\npublic inline fun <K, V, M : MutableMap<K, MutableList<V>>, UByteArray, UShortArray> groupByTo(destination: M, keySelector: (UByte) -> K, valueTransform: (UByte) -> V): M {
    for (element in this) {
        val key = keySelector(element)\n        val list = destination.getOrPut(key) { ArrayList<V>() }\n        list.add(valueTransform(element))\n    }\n    return destination\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.mapIndexed(transform: (index: Int, UByte) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.mapIndexed(transform: (Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}

/**
 * Returns a list containing the results of applying the given [transform] function\n* to each element and its index in the original array.\n* @param 
* [transform] function that takes an element and returns the result of the transform applied to the element.\n* @sample samples.collections.Collections.Transformations.mapIndexedTo
*/
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.mapIndexed(transform: (Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)\n}
Iterable<IndexedValue<UInt>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the
 * original array into an [IndexedValue] containing the index of
 * that element and the element itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.withIndex():
    Iterable<IndexedValue<ULong>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns a lazy [Iterable]
 * that wraps each element of the original array into an [IndexedValue] containing the index of that element and the
 * element itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.withIndex():
    Iterable<IndexedValue<UByte>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns a lazy [Iterable]
 * that wraps each element of the original array into an [IndexedValue] containing the index of that element and the
 * element itself.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.withIndex():
    Iterable<IndexedValue<UShort>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.all(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.any(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.any(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.any(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.any(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
@sample samples.collections.Collections.Aggregates.anyWithPredicate

/**
 * Returns `true` if at least one element matches the given [predicate].
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.any(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.any(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.count(predicate: (UInt) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.count(predicate: (ULong) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.count(predicate: (UByte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.count(predicate: (UShort) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 * @param [operation] function that takes the index of an element, current accumulator value\n * and the element itself, and calculates the next accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}
right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value
* and the element itself, and calculates the next accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldIndexed(initial: R, operation: (Int, acc: R, UByte) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldIndexed(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.foldIndexed(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (Int, acc: R, UByte) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldRightIndexed(initial: R, operation: (UInt, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.foldRightIndexed(initial: R, operation: (ULong, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (Int, acc: R, UByte) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right to left to each element with its index in the original array and current accumulator value.\n
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
value and applying [operation] from right to left\n* to each element with its index in the original array and current
accumulator value.\n* @param [operation] function that takes the index of an element, the element itself\n* and current accumulator value, and calculates the next accumulator value.\n
```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (index: Int, UByte, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.forEach(action: (UInt) -> Unit): Unit {
    for (element in this) action(element)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.forEach(action: (ULong) -> Unit): Unit {
    for (element in this) action(element)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.forEach(action: (UByte) -> Unit): Unit {
    for (element in this) action(element)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.forEach(action: (UShort) -> Unit): Unit {
    for (element in this) action(element)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.forEachIndexed(action: (index: Int, UInt) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.forEachIndexed(action: (index: Int, ULong) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.forEachIndexed(action: (index: Int, UByte) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.forEachIndexed(action: (index: Int, UShort) -> Unit): Unit {
    var index = 0
    for (item in this) action(index++, item)
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.max(): UInt? {
    if (isEmpty())
        return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e) max = e
    return max
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.max(): UByte? {
    if (isEmpty())
        return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e) max = e
    return max
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.max(): UShort? {
    if (isEmpty())
        return null
    var max = this[0]
    for (i in 1..lastIndex)
        if (max < e) max = e
    return max
}
```
fun ULongArray.max(): ULong? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

fun UByteArray.max(): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

fun UShortArray.max(): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (max < e) max = e
    }
    return max
}

fun UIntArray.maxBy(selector: (UInt) -> R): UInt? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

fun ULongArray.maxBy(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

fun UByteArray.maxBy(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

fun UShortArray.maxBy(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

fun UIntArray.maxWith(comparator: Comparator<in UInt>): UInt? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun ULongArray.maxWith(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UByteArray.maxWith(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}
first element having the largest value according to the provided [comparator] or `null` if there are no elements.

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UShortArray.maxWith(comparator: Comparator<in UShort>): UShort? {\n    \n    if (isNullOrEmpty()) return null
    \n    var min = this[0] for (i in 1..lastIndex) {\n        val e = this[i]
        \n        if (comparator.compare(min, e) > 0) min = e
    }
    \n    return min
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UIntArray.min(): UInt? {
    \n    if (isNullOrEmpty()) return null
    \n    var min = this[0] for (i in 1..lastIndex) {\n        val e = this[i]
        \n        if (min > e) min = e
    }
    \n    return min
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun ULongArray.min(): ULong? {
    \n    if (isNullOrEmpty()) return null
    \n    var min = this[0] for (i in 1..lastIndex) {\n        val e = this[i]
        \n        if (min > e) min = e
    }
    \n    return min
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UByteArray.min(): UByte? {
    \n    if (isNullOrEmpty()) return null
    \n    var min = this[0] for (i in 1..lastIndex) {\n        val e = this[i]
        \n        if (min > e) min = e
    }
    \n    return min
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UShortArray.min(): UShort? {
    \n    if (isNullOrEmpty()) return null
    \n    var min = this[0] for (i in 1..lastIndex) {\n        val e = this[i]
        \n        if (min > e) min = e
    }
    \n    return min
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UIntArray.minBy(selector: (UInt) -> R): UInt? {
    \n    if (isNullOrEmpty()) return null
    \n    var minElem = this[0]
    \n    val lastIndex = this.lastIndex
    \n    if (lastIndex == 0) return minElem
    \n    var minValue = selector(minElem)
    \n    for (i in 1..lastIndex) {\n        val e = this[i]
        \n        val v = selector(e)
        \n        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    \n    return minElem
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun ULongArray.minBy(selector: (ULong) -> R): ULong? {
    \n    if (isNullOrEmpty()) return null
    \n    var minElem = this[0]
    \n    val lastIndex = this.lastIndex
    \n    if (lastIndex == 0) return minElem
    \n    var minValue = selector(minElem)
    \n    for (i in 1..lastIndex) {\n        val e = this[i]
        \n        val v = selector(e)
        \n        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    \n    return minElem
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UByteArray.minBy(selector: (UByte) -> R): UByte? {
    \n    if (isNullOrEmpty()) return null
    \n    var minElem = this[0]
    \n    val lastIndex = this.lastIndex
    \n    if (lastIndex == 0) return minElem
    \n    var minValue = selector(minElem)
    \n    for (i in 1..lastIndex) {\n        val e = this[i]
        \n        val v = selector(e)
        \n        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    \n    return minElem
}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*\@public fun UShortArray.minBy(selector: (UShort) -> R): UShort? {
    \n    if (isNullOrEmpty()) return null
    \n    var minElem = this[0]
    \n    val lastIndex = this.lastIndex
    \n    if (lastIndex == 0) return minElem
    \n    var minValue = selector(minElem)
    \n    for (i in 1..lastIndex) {\n        val e = this[i]
        \n        val v = selector(e)
        \n        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    \n    return minElem
}
element having the smallest value according to the provided [comparator] or `null` if there are no elements.

```kotlin
*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\npublic fun ULongArray.minWith(comparator: Comparator<in ULong>): ULong? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\npublic fun UByteArray.minWith(comparator: Comparator<in UByte>): UByte? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\npublic fun UShortArray.minWith(comparator: Comparator<in UShort>): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```kotlin
/**
 * Returns `true` if the array has no elements.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.none(): Boolean {
    return isEmpty()
}
```

```kotlin
/**
 * Returns `true` if the array has no elements.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.none(): Boolean {
    return isEmpty()
}
```

```kotlin
/**
 * Returns `true` if the array has no elements.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.none(): Boolean {
    return isEmpty()
}
```

```kotlin
/**
 * Returns `true` if no elements match the given [predicate].

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.none(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```kotlin
/**
 * Returns `true` if no elements match the given [predicate].

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.none(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```kotlin
/**
 * Returns `true` if no elements match the given [predicate].

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.none(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```kotlin
/**
 * Returns `true` if no elements match the given [predicate].

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.none(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

*\n@SinceKotlin("1.3\")@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UShortArray.reduce(operation: (acc: UShort, UShort) -> UShort): UShort {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right
 * to current accumulator value and each element
 * @param [operation] function that takes an index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
accumulator

 Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the array is empty.

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceOrNull(operation: (acc: UInt, UInt) -> UInt): UInt? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceOrNull(operation: (acc: ULong, ULong) -> ULong): ULong? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRight(operation: (ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

 ```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort) : UShort {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UIntArray.reduceRightIndexed(operation: (Int, UInt, acc: UInt) -> UInt) : UInt {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULongArray.reduceRightIndexed(operation: (Int, ULong, acc: ULong) -> ULong) : ULong {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByteArray.reduceRightIndexed(operation: (Int, UByte, acc: UByte) -> UByte) : UByte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value. Returns null if the array is empty.
 * @sample samples.collections.Collections.Aggregates.reduceRightOrNull
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UIntArray.reduceRightOrNull(operation: (UInt, acc: UInt) -> UInt) : UInt? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ULongArray.reduceRightOrNull(operation: (Int, ULong, acc: ULong) -> ULong) : ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun UByteArray.reduceRightOrNull(operation: (Int, UByte, acc: UByte) -> UByte) : UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal:inline
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
        --index
    }
    return accumulator
}
public inline fun ULongArray.reduceRightOrNull(operation: (ULong, acc: ULong) -> ULong): ULong? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value.
 * Returns null if the array is empty.
 * @sample
 * samples.collections.Collections.Aggregates.reduceRightOrNull
 */
public inline fun UByteArray.reduceRightOrNull(operation: (UByte, acc: UByte) -> UByte): UByte? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from right to left.
 * @param
 * @sample
 * samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> UIntArray.scan(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * @param
 * @sample
 * samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> ULongArray.scan(initial: R, operation: (acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right.
 * @param
 * @sample
 * samples.collections.Collections.Aggregates.scan
 */
public inline fun <R> UByteArray.scan(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
[operation] from left to right * to each element and current accumulator value that starts with [initial] value.

* Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

* @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

* @sample samples.collections.Collections.Aggregates.scan

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inlineOn
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this)
        accumulator = operation(accumulator, element)
    result.add(accumulator)
    return result
}

/*
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * 
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inlineOn
public inline fun <R> UIntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices)
        accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
    return result
}

/*
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * 
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inlineOn
public inline fun <R> ULongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices)
        accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
    return result
}

/*
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * 
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inlineOn
public inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices)
        accumulator = operation(index, accumulator, this[index])
    result.add(accumulator)
    return result
}
```

---

Open Source Used In DNAC Platform Arica 13272
accumulator = initial
for (index in indices) {
    accumulator = operation(index, accumulator, this[index])
} return result

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right on each element and current accumulator value that starts with the first element of this array.
 * Note that the `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * This function takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.scanReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right on each element and current accumulator value that starts with the first element of this array.
 * Note that the `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * This function takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.scanReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right on each element and current accumulator value that starts with the first element of this array.
 * Note that the `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * This function takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.scanReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right on each element and current accumulator value that starts with the first element of this array.
 * Note that the `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * This function takes current accumulator value and an element, and calculates the next accumulator value.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
public inline fun UIntArray.scanReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to
 * each element, its index in the original array and current accumulator value that starts with the first element of this
 * array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the
 * previous value in resulting list. The value passed to the index element, current accumulator value and the element
 * itself, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
public inline fun ULongArray.scanReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to
 * each element, its index in the original array and current accumulator value that starts with the first element of this
 * array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the
 * previous value in resulting list. The value passed to the index element, current accumulator value and the element
 * itself, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
public inline fun UByteArray.scanReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to
 * each element, its index in the original array and current accumulator value that starts with the first element of this
 * array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the
 * previous value in resulting list. The value passed to the index element, current accumulator value and the element
 * itself, and calculates the next accumulator value.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun ULongArray.sumBy(selector: (ULong) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun UByteArray.sumBy(selector: (UByte) -> UByte): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the array.
 */
public inline fun UShortArray.sumBy(selector: (UShort) -> UShort): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.scanReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.scanReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.scanReduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumBy(selector: (ULong) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumBy(selector: (UByte) -> UByte): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/*@SinceKotlin("1.3")*/
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumBy(selector: (UShort) -> UShort): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
UByteArray.sumBy(selector: (UByte) -> UInt): UInt {
    var sum = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.sumBy(selector: (UByte) -> UInt): UInt
    = UByteArray.sumByDouble(selector = selector)

UShortArray.sumBy(selector: (UShort) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.sumByDouble(selector: (UShort) -> Double): Double
    = UShortArray.sumByDouble(selector = selector)

UIntArray.sumByDouble(selector: (UInt) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.sumByDouble(selector: (UInt) -> Double): Double
    = UIntArray.sumByDouble(selector = selector)

ULongArray.sumByDouble(selector: (ULong) -> Double): Double {
    var sum = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector]
 * function applied to each element in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.sumByDouble(selector: (ULong) -> Double): Double
    = ULongArray.sumByDouble(selector = selector)

/**
 * Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>>
    = UIntArray.zip(other)

/**
 * Returns a list of pairs built from the elements of 'this' array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> ULongArray.zip(other: Array<out R>): List<Pair<ULong, R>>
    = ULongArray.zip(other)

/**
 * Returns a list of values built from the elements of 'this' array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V> UIntArray.zip(other: Array<out R>, transform: (a: UInt, b: R) -> V): List<V>
    = UIntArray.zip(other, transform)

/**
 * Returns a list of values built from the elements of 'this' array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V>
    = ULongArray.zip(other, transform)

Open Source Used In DNAC Platform Arica 13275
public inline fun <R, V> ULongArray.zip(other: Array<out R>, transform: (a: ULong, b: R) -> V): List<V> {
    var size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */

public inline fun <R, V> UByteArray.zip(other: Array<out R>, transform: (a: UByte, b: R) -> V): List<V> {
    var size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array
 * using the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */

    var size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.
 * The returned list has length of the shortest collection.
 */

public infix fun <R> UIntArray.zip(other: Iterable<R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs
 * built from the elements of `this` collection and [other] array with the same index.
 * The returned list has length of
 * the shortest collection.
 */

public infix fun <R> ULongArray.zip(other: Iterable<R>): List<Pair<ULong, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs
 * built from the elements of `this` collection and [other] array with the same index.
 * The returned list has length of
 * the shortest collection.
 */

public infix fun <R> UByteArray.zip(other: Iterable<R>): List<Pair<UByte, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs
 * built from the elements of `this` collection and [other] array with the same index.
 * The returned list has length of
 * the shortest collection.
 */

public infix fun <R> UShortArray.zip(other: Iterable<R>): List<Pair<UShort, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
Returns a list of values built from the elements of `this` array and the [other] collection with the same index
using the provided [transform] function applied to each pair of elements. The returned list has length of the shortest
collection. * @sample samples.collections.Iterables.Operations.zipIterableWithTransform*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R, V> UByteArrayList.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<V> {\n    val arraySize = size* val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other) {\n        if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list\n} *\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <R, V> UShortArrayList.zip(other: Iterable<R>, transform: (a: UShort, b: R) -> V): List<V> {\n    val arraySize = size* val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))\n    var i = 0\n    for (element in other) {\n        if (i >= arraySize) break\n        list.add(transform(this[i++], element))\n    }\n    return list\n}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection. * @sample samples.collections.Iterables.Operations.zipIterable*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun UIntArrayList.zip(other: UIntArrayList): List<Pair<UInt, UInt>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n} *\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun ULongArrayList.zip(other: ULongArrayList): List<Pair<ULong, ULong>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection. * @sample samples.collections.Iterables.Operations.zipIterable*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun UByteArray.zip(other: UByteArray): List<Pair<UByte, UByte>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection. * @sample samples.collections.Iterables.Operations.zipIterable*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun UShortArrayList.zip(other: UShortArrayList): List<Pair<UShort, UShort>> {\n    return zip(other) { t1, t2 -> t1 to t2 }\n}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements. The returned list has length of the shortest
collection. * @sample samples.collections.Iterables.Operations.zipIterableWithTransform*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <V> UIntArrayList.zip(other: UIntArrayList, transform: (a: UInt, b: UInt) -> V): List<V> {\n    val size = minOf(size, other.size)* val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements. The returned list has length of the shortest
collection. * @sample samples.collections.Iterables.Operations.zipIterableWithTransform*

*\@SinceKotlin("1.3.0")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\@public inline fun <V> ULongArrayList.zip(other: ULongArrayList, transform: (a: ULong, b: ULong) -> V): List<V> {\n    val size = minOf(size, other.size)* val list = ArrayList<V>(size)\n    for (i in 0 until size) {\n        list.add(transform(this[i], other[i]))\n    }\n    return list\n}
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Array<out ULong>.sum(): ULong {
    var sum: ULong = 0UL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.sum(): UInt {
    return storage.sum().toUInt()
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.sum(): ULong {
    return storage.sum().toULong()
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.sum(): UInt {
    return sumBy { it.toUInt() }
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.sum(): UInt {
    return sumBy { it.toUInt() }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UCollectionsKt")

package
kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
import
kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an array of UByte containing all of the elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun
Collection<UByte>.toUByteArray(): UByteArray {
    val result = UByteArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns an array of UShort containing all of the elements of this collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun
Collection<UShort>.toUShortArray(): UShortArray {
    val result = UShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 */
@kotlin.jvm.JvmName("zipIterableWithTransform")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
Collection<out UByte>.zip(other: UByteArray, transform: (a: UByte, b: UByte) -> Int): List<Int> {
    val size = minOf(size, other.size)
    val list = ArrayList<Int>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns a list of values built from the elements of 'this' array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 */
@kotlin.jvm.JvmName("zipIterableWithTransform")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun
    val size = minOf(size, other.size)
    val list = ArrayList<Int>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
UShortArray {
    val result = UShortArray(size)
    var index = 0
    for (element in this)
        result[index++] = element
    return result
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<ULong>.sum(): ULong {
    var sum: ULong = 0UL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Iterable<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * 
 * @file:kotlin.jvm.JvmMultifileClass
 * @file:kotlin.jvm.JvmName("UComparisonsKt")
 * package kotlin.comparisons
 * 
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UInt, b: UInt): UInt {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: ULong, b: ULong): ULong {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UByte, b: UByte): UByte {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun maxOf(a: UShort, b: UShort): UShort {
    return if (a >= b) a else b
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UByte, b: UByte, c: UByte): UByte {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return maxOf(a, maxOf(b, c))
}

/*
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UInt, b: UInt): UInt {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UByte, b: UByte): UByte {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun minOf(a: UShort, b: UShort): UShort {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UInt, b: UInt, c: UInt): UInt {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: ULong, b: ULong, c: ULong): ULong {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return minOf(a, minOf(b, c))
}
```kotlin
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline fun minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
    return minOf(a, minOf(b, c))
}

/*
 * Returns a random element from this range. * @throws IllegalArgumentException if this range is empty.*
*/
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline fun UIntRange.random(): UInt {
    return random(random(Random))
}

/**
 * Returns a random element from this range. * @throws IllegalArgumentException if this range is empty.*
*/
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public inline fun ULongRange.random(): ULong {
    return random(random(Random))
}

/**
 * Returns a random element from this range using the specified source of randomness, or 'null' if this range is empty.*
*/
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntRange.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return random.nextUInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or 'null' if this range is empty.*
*/
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongRange.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return random.nextULong(this)
}

/**
 * Checks if the specified [element] belongs to this range.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntRange.contains(value: UByte): Boolean {
    return contains(value.toUInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongRange.contains(value: UByte): Boolean {
    return contains(value.toULong())
}
```

---

Open Source Used In DNAC Platform Arica 13280
specified [value] belongs to this range.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: ULong): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: ULong): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: ULong): Boolean {
    return (value shr UInt.SIZE_BITS) == 0uL && contains(value.toUInt())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UIntRange.contains(value: UShort): Boolean {
    return contains(value.toUInt())
}
```

```kotlin
/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULongRange.contains(value: UShort): Boolean {
    return contains(value.toULong())
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)
}
```

```kotlin
/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * If the [to] value is greater than `this` value the returned progression is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULong.downTo(to: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(this, to, -1)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range in the opposite direction with the step -1.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)
}
```

```kotlin
/**
 * Returns a progression that goes over the same range with the given step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UIntProgression.step(step: Int): UIntProgression {
    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UByte.until(to: UByte): UIntRange {
    if (to <= UByte.MIN_VALUE) return UIntRange.EMPTY
    return this.toUInt() .. (to - 1u).toUInt()
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UInt.until(to: UInt): UIntRange {
    if (to <= UInt.MIN_VALUE) return UIntRange.EMPTY
    return this .. (to - 1u).toUInt()
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULong.until(to: ULong): ULongRange {
    if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY
    return this .. (to - 1L).toULong()
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UShort.until(to: UShort): UIntRange {
    return UIntRange.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UShort.until(to: UShort): UIntRange {
    return UIntRange.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

```kotlin
/**
 * Returns a range from this value up to but excluding the specified [to] value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongRange.contains(value: UShort): Boolean {
    return contains(value.toULong())
}
```
infix fun ULong.until(to: ULong): ULongRange {
    if (to <= ULong.MIN_VALUE) return ULongRange.EMPTY
    return this .. (to - 1u).toULong()  // Ensures that this value is not less than the specified [minimumValue].
}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt) : UInt {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

// Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(range: ClosedRange<UInt>) : UInt {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<UInt>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong) : ULong {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(range: ClosedRange<ULong>) : ULong {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<ULong>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte) : UByte {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(range: ClosedRange<UByte>) : UByte {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<UByte>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort) : UShort {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(range: ClosedRange<UShort>) : UShort {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<UShort>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("USequencesKt")

package kotlin.sequences

// See:
import kotlin.random.*

/**
 * Returns the sum of all elements in the sequence. The operation is _terminal_
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence. The operation is _terminal_
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence. The operation is _terminal_
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence. The operation is _terminal_
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License,
 * Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package kotlin.collections

import kotlin.internal.PlatformDependent

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that can\n * be iterated over. @param T the type of element being iterated over. The iterator is covariant on its element type.
 * @param T the type of element being iterated over. The iterator is covariant on its element type.
 */
public interface Iterable<out T> {
    /**
     * Returns an iterator over the elements of this object.
     */
    public operator fun iterator(): Iterator<T>
}

/**
 * Classes that inherit from this interface can be represented as a sequence of elements that can\n * be iterated over and that supports removing elements during iteration. @param T the type of element being iterated over.
 * The mutable iterator is invariant on its element type.
 */
public interface MutableIterable<out T> : Iterable<T> {
    /**
     * Returns an iterator over the elements of this sequence that supports removing elements during iteration.
     */
    override fun iterator(): MutableIterator<T>
}

/**
 * A generic collection of elements. Methods in this interface support only read-only access to the collection. Read/write access is supported through the
 * [MutableCollection] interface.
 * @param E the type of elements contained in the collection. The collection is covariant on its element type.
 */
public interface Collection<out E> : Iterable<E> {
    // Query Operations
    /**
     * Returns the size of the collection.
     */
    public val size: Int

    /**
     * Returns `true` if the collection is empty (contains no elements), `false` otherwise.
     */
    public fun isEmpty(): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public operator fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operations
    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean
}

/**
 * A generic collection of elements that supports adding and removing elements. Methods in this interface support only read-only access to the collection. Read/write access is supported through the
 * [MutableCollection] interface.
 * @param E the type of elements contained in the collection. The collection is covariant on its element type.
 */
public interface MutableCollection<E> : Collection<E>, MutableIterable<E> {
    // Query Operations
    /**
     * Returns the size of the collection.
     */
    public val size: Int

    /**
     * Returns 'true' if the collection is empty (contains no elements), 'false' otherwise.
     */
    public fun isEmpty(): Boolean

    /**
     * Checks if the specified element is contained in this collection.
     */
    public fun contains(element: @UnsafeVariance E): Boolean

    override fun iterator(): Iterator<E>

    // Modification Operations
    /**
     * Checks if all elements in the specified collection are contained in this collection.
     */
    public fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean

    override fun add(element: E): Boolean

    override fun remove(element: E): Boolean

    // Bulk Operation
    /**
     * Adds the specified element to the collection. The collection does not support duplicates\n     * and the element is already contained in the collection.
     */
    public fun add(element: E): Boolean

    /**
     * Removes a single instance of the specified element from this collection, if it is present.
     */
    public fun remove(element: E): Boolean

    /**
     * Adds all of the elements of the specified collection to this collection.
     */
    public fun addAll(elements: Collection<@UnsafeVariance E>): Boolean

    override fun iterator(): Iterator<E>

    // Bulk Operation
    /**
     * Adds all of the elements of the specified collection to this collection. If any of the specified elements was added to the collection, 'false' if the
collection was not modified.
 */
 public fun addAll(elements: Collection<E>): Boolean
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Search Operations
 */
 * Returns the index of the first occurrence of the specified element in the list, or -1 if the specified element is not contained in the list.
 */
 public fun indexOf(element: @UnsafeVariance E): Int

 */
 * Returns the index of the last occurrence of the specified element in the list, or -1 if the specified element is not contained in the list.
 */
 public fun lastIndexOf(element: @UnsafeVariance E): Int

 // List Iterators
 */
 * Returns a list iterator over the elements in this list (in proper sequence).
 */
 public fun listIterator(): ListIterator<E>

 // View
 */
 * Returns a view of the portion of this list between the specified [fromIndex] (inclusive) and [toIndex] (exclusive).
 */
 public fun subList(fromIndex: Int, toIndex: Int): List<E>

 // Iterator
 */
 * Returns an iterator over the elements in this list (in proper sequence), starting at the specified [index].
 */
 public fun iterator(): Iterator<E>

 // Bulk Operations
 */
 * Adds all of the elements of the specified collection to the end of this list.
 */
 public fun addAll(elements: Collection<E>): Boolean

 */
 * Inserts all of the elements of the specified collection into this list at the specified [index].
 */
 public fun addAll(index: Int, elements: Collection<E>): Boolean

 */
 * Removes all elements from this collection.
 */
 public fun clear(): Unit

 */
 * Removes all elements from this collection.
 */
 public fun removeAll(elements: Collection<E>): Boolean

 */
 * Retains only the elements in this collection that are contained in the specified collection.
 */
 public fun retainAll(elements: Collection<E>): Boolean

 */
 * Removes all elements from this collection.
 */
 public fun removeAll(elements: Collection<E>): Boolean

 */
 * Retains only the elements in this collection that are contained in the specified collection.
 */
 public fun retainAll(elements: Collection<E>): Boolean

 // Positional Access Operations
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun set(index: Int, element: E): E

 // Positional Access
 */
 * A generic ordered collection of elements.
 */
 public interface Collection<E>: Collection<@UnsafeVariance E>

 // Positional Access Operations
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
 */
 * Returns the element at the specified index in the list.
 */
 public operator fun get(index: Int): E

 // Positional Access
may contain duplicate values.

override val values: MutableCollection<V>

/*\n
Returns a [MutableSet] of all key/value pairs in this map.

*\n
override val entries:

MutableSet<MutableMap.Entry<K, V>>\n
*\n
Represents a key/value pair held by a [MutableMap].

*/\n
public interface MutableEntry<K, V>: Map.Entry<K, V>

/*\n
Changes the value associated with the key of this entry.

*/\n
@volatile

public fun setValue(newValue: V): V

*/

public interface MutableEntry<K, V>: Map.Entry<K, V>

/*\n
Changes the value associated with the key of this entry.

*/\n
@volatile

public fun setValue(newValue: V): V

*/
throw kotlin.NoSuchElementException().

throw kotlin.NoSuchElementException().
LongProgressionIterator(first, last, step)  /** Checks if the progression is empty. */

    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
      other is LongProgression && (isEmpty() && other.isEmpty() ||
        first == other.first && last == other.last &&
        step == other.step)

    override fun hashCode(): Int =
      if (isEmpty()) -1 else 31 * (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))) + (step xor (step ushr 32)).toInt()

    override fun toString(): String = if (step > 0)
      "$first..$last step $step" else "$first downTo $last step ${-step}"

    companion object {
      /**
       * Creates LongProgression within the specified bounds of a closed range.
       * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step]. In order to go backwards the [step] must be negative. *
       * [step] must be greater than `Long.MIN_VALUE` and not equal to zero. *
       * public fun fromClosedRange(rangeStart: Long, rangeEnd: Long, step: Long): LongProgression = LongProgression(rangeStart, rangeEnd, step)
       */
    }
}

irtual interface ClosedRange<T: Comparable<T>> {
    /**
     * The minimum value in the range. *
     */
    public val start: T

    /**
     * The maximum value in the range (inclusive). *
     */
    public val endInclusive: T

    /**
     * Checks whether the specified [value] belongs to the range. *
     */
    operator fun contains(value: T): Boolean = value >= start && value <= endInclusive

    /**
     * Checks whether the range is empty. *
     */
    public fun isEmpty(): Boolean = start > endInclusive
}

package kotlin

/**
 * The type with only one value: the `Unit` object. This type corresponds to the `void` type in Java. *
 * Contains the list of code elements which are the possible annotation targets *
 * public enum class AnnotationTarget {
 *   CLASS, *
 *   ANNOTATION_CLASS, *
 *   TYPE_PARAMETER, *
 *   PROPERTY, *
 *   LOCAL_VARIABLE, *
 *   VALUE_PARAMETER, *
 *   CONSTRUCTOR, *
 *   Function (constructors are not included) *
 *   PROPERTY_GETTER, *
 *   PROPERTY_SETTER, *
 *   TYPE, *
 *   Any expression *
 *   File *
 * @SinceKotlin("1.1")
 * Contains the list of possible annotation's retentions. *
 * Determines how an annotation is stored in binary output. *
 * Annotation isn't stored in binary output *
 * Annotations is stored in binary output, but invisible for reflection *
 * Annotations is stored in binary output and visible for reflection (default retention) *
 */

possible targets of an annotation.

* If the target meta-annotation is not present on an annotation declaration, the annotation is applicable to the following elements:
  * [CLASS], [PROPERTY], [FIELD], [LOCAL_VARIABLE], [VALUE_PARAMETER], [CONSTRUCTOR], [FUNCTION], [PROPERTY_GETTER], [PROPERTY_SETTER].

* @property allowedTargets list of allowed annotation targets

* @Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
@Retention(AnnotationRetention.RUNTIME, BINARY or SOURCE)
public annotation class Target(vararg val allowedTargets: AnnotationTarget)

* @Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Retention(val value: AnnotationRetention = AnnotationRetention.RUNTIME)

* @Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class Repeatable

* @Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class MustBeDocumented

---

/*
 * Copyright 2010-2016 JetBrains s.r.o.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.internal

/**
 * Specifies that the corresponding type parameter is not used for unsafe operations such as casts or 'is' checks
 * That means it's completely safe to use generic types as argument for such parameter.
 *
 * @Target(AnnotationTarget.TYPE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
internal annotation class PureReifiable

/**
 * Specifies that the corresponding built-in method exists depending on platform.
 * Current implementation for JVM looks whether method with same JVM descriptor exists in the module JDK.
 * For example MutableMap.remove(K, V) available only if corresponding method
 * 'java/util/Map.remove(Ljava/lang/Object;Ljava/lang/Object;)Z' is defined in JDK (i.e. for major versions >= 8)
 *
 * @Target(AnnotationTarget.FUNCTION)
@Retention(AnnotationRetention.BINARY)
internal annotation class PlatformDependent
*/

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.internal

// a mod b (in arithmetical sense)
private fun mod(a: Int, b: Int): Int = a % b
private fun mod(a: Long, b: Long): Long = a % b

// (a - b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int = mod(mod(a, c) - mod(b, c), c)
private fun differenceModulo(a: Long, b: Long, c: Long): Long = mod(mod(a, c) - mod(b, c), c)

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range of [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 * No validation on passed parameters is performed. The given parameters should satisfy the condition:
 * - either `step > 0` and `start <= end`,
 * - or `step < 0` and `start >= end`.
 * @param start first element of the progression
 * @param end ending bound for the progression
 * @param step step increment, or difference of successive elements in the progression
 * @return the final element of the progression
 */
@Suppress("*in*PublishedApi*

fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start <= end) end else differenceModulo(start, end, step)
    step < 0 -> if (start < end) end else differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start <= end) end else differenceModulo(start, end, step)
    step < 0 -> if (start < end) end else differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start <= end) end else differenceModulo(start, end, step)
    step < 0 -> if (start < end) end else differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start <= end) end else differenceModulo(start, end, step)
    step < 0 -> if (start < end) end else differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}
satisfy the condition:
- either `step > 0` and `start <= end`,
- or `step < 0` and `start >= end`.

@param start first element of the progression
@param end ending bound for the progression
@suppress

@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long):
    Long = when {
        step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
        step < 0 -> if (start <= end) end else end + differenceModulo(start, end, step)
        else -> throw kotlin.IllegalArgumentException("Step is zero.")
    }

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under
 * the License.
 */

package kotlin.reflect

/**
 * Represents a type. Type is usually either a class with optional type arguments,
 * or a type parameter of some declaration, plus nullability.
 */
public interface KType : KAnnotatedElement {
    /**
     * The declaration of the classifier used in this type.
     * For example, in the type `List<String>` the classifier would be the [KClass]
     * instance for [List].
     */
    @SinceKotlin("1.1.1")
    public val classifier: KClassifier?

    /**
     * Type arguments passed for the parameters of the classifier in this type.
     * The declaration of the classifier used in this type.
     */
    @SinceKotlin("1.1.1")
    public val arguments: List<KTypeProjection>

    /**
     * `true` if this parameter is optional and can be omitted when making a call via [KCallable.callBy],
     * or `false` otherwise.
     */
    @SinceKotlin("1.1")
    public val isOptional: Boolean

    /**
     * Returns `null` if this type is not denotable in Kotlin, for example if it is an intersection type.
     */
    @sinceKotlin("1.1.1")
    public val isNullable: Boolean

    /**
     * `true` if this parameter is `vararg`.
     */
    @SinceKotlin("1.1")
    public val isVararg: Boolean

    /**
     * In case this type is based on an inner class, the
     * returned list contains the type arguments provided for the innermost class first,
     * then its outer class, and so on.
     */
    @SinceKotlin("1.1.1")
    public val arguments: List<KTypeProjection>

    /**
     * `true` if this was declared with a question mark at the end.
     * For non-Kotlin types, it means the type or the symbol which was declared with this type
     * is annotated with a
runtime-retained nullability annotation such as `@javax.annotation.Nullable`. This may happen if it is a type of the type parameter with a nullable upper bound. The use-site variance specified in the projection, or `null` if this is a star projection. Without any use-site variance modifiers applied to it. For example, in the type `MutableList<in Number>` is a contravariant projection of the type of class `Number`. `out Number` is a covariant projection of the type denoted by the `out` modifier applied to a type. For example, in the type `Array<out Number>`, `out Number` is a covariant projection of the type represented by the class `Number`.
Java visibilities such as package-private and protected (which also gives access to items from the same package) cannot be represented in Kotlin, so there's no [KVisibility] value corresponding to them. See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/visibility-modifiers.html) for more information.

```kotlin
@SinceKotlin("1.1")
enum class KVisibility {
    /**
     * Visibility of declarations marked with the `public` modifier, or with no modifier at all.
     */
    PUBLIC,

    /**
     * Visibility of declarations marked with the `protected` modifier.
     */
    PROTECTED,

    /**
     * Visibility of declarations marked with the `package-private` modifier.
     */
    INTERNAL,

    /**
     * Visibility of declarations marked with the `private` modifier.
     */
    PRIVATE,
}

* * *

See the [Kotlin language documentation](https://kotlinlang.org/docs/reference/visibility-modifiers.html) for more information.

```
if (js("Error").captureStackTrace) {
    js("Error").captureStackTrace(instance, instance::class.js);
} else {
    instance.asDynamic().stack = js("new Error()").stack;
}

internal fun newThrowable(message: String?, cause: Throwable?): Throwable {
    val throwable = js("new Error()")
    throwable.message = if (jsTypeOf(message) == "undefined") {
        if (cause != null) cause.toString() else null
    } else {
        message
    }
    throwable.cause = cause
    throwable.name = "Throwable";
    return throwable
}

internal fun boxedChar(c: Int) = BoxedChar(c)

internal fun emptyArray(): Array<*> = js("[]")

internal fun arrayConcat<T>(a: T, b: T): T = concat(js("arguments"))
encapsulating it as a failure.

```kotlin
T.runCatching(block: T.() -> R): Result<T, R> {
    return try {
        block()
    } catch (e: Throwable) {
        Result.failure(e)
    }
}
```

```kotlin
public inline fun <T, R> T.runCatching(block: T.() -> R): Result<R> {
    return try {
        Result.success(block())
    } catch (e: Throwable) {
        Result.failure(e)
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.getOrThrow(): T {
    throwOnFailure()
    return value as T
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.getOrElse(onFailure: (exception: Throwable) -> R): R {
    contract {
        callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> value as T
        else -> onFailure(exception)
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recover(onFailure: (exception: Throwable) -> R): Result<R> {
    contract {
        callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(onFailure(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { onFailure(exception) }
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.getOrElse(defaultValue: R): R {
    if (isFailure) return defaultValue
    return value as T
}
```

```kotlin
/**
 * Returns the encapsulated value if this instance represents 
 * [success][Result.isSuccess] or throws the encapsulated [Throwable] exception if it is 
 * [failure][Result.isFailure]. *
 * Note, that this function rethrows any [Throwable] exception thrown by 
 * [onFailure] function. *
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { onFailure(exception) }
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.fold(onSuccess: (value: T) -> R, onFailure: (exception: Throwable) -> R): R {
    contract {
        callsInPlace(onSuccess, InvocationKind.AT_MOST_ONCE)
        callsInPlace(onFailure, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> onSuccess(value as T)
        else -> onFailure(exception)
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.mapCatching(transform: (value: T) -> R): Result<R> {
    return when {
        isSuccess -> runCatching { transform(value as T) }
        else -> Result(value)
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.recoverCatching(onFailure: (exception: Throwable) -> R): Result<R> {
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}
```
or the original encapsulated value if it is [success][Result.isSuccess].
*\n* This function catches any [Throwable] exception thrown by [transform] function and encapsulates it as a failure.
*\n* See [recover] for an alternative that rethrows exceptions.

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R>
Result<T>.recoverCatching(transform: (exception: Throwable) -> R): Result<R> {
    val value = value // workaround for inline classes BE bug
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { transform(exception) }
    }
}
```

```kotlin
// "peek" onto value/exception and pipe

/**
 * Performs the given [action] on the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    exceptionOrNull()?.let { action(it) }
    return this
}
```

```kotlin
/**
 * Performs the given [action] on the encapsulated value if this instance represents [success][Result.isSuccess].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    if (isSuccess) action(value as T)
    return this
}
```

```kotlin
// -------------------

```

---

or the original encapsulated value if it is [success][Result.isSuccess].
*\n* This function catches any [Throwable] exception thrown by [transform] function and encapsulates it as a failure.
*\n* See [recover] for an alternative that rethrows exceptions.

```kotlin
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R>
Result<T>.recoverCatching(transform: (exception: Throwable) -> R): Result<R> {
    val value = value // workaround for inline classes BE bug
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { transform(exception) }
    }
}
```

```kotlin
// "peek" onto value/exception and pipe

/**
 * Performs the given [action] on the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    exceptionOrNull()?.let { action(it) }
    return this
}
```

```kotlin
/**
 * Performs the given [action] on the encapsulated value if this instance represents [success][Result.isSuccess].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {
    contract {
        callsInPlace(action, InvocationKind.AT_MOST_ONCE)
    }
    if (isSuccess) action(value as T)
    return this
}
```

```kotlin
// -------------------

```
continuation will produce an [IllegalStateException].

*\@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun <R, T> (suspend R.() -> T).createCoroutine(
    receiver: R, \n    completion: Continuation<T>\n): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(),
    COROUTINE_SUSPENDED)\n
*\@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun <R, T> (suspend R.() -> T).startCoroutine(
    receiver: R, \n    completion: Continuation<T>\n): Continuation<Unit> =
    createCoroutineUnintercepted(receiver, completion).intercepted().resume(Unit)\n
*\@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun <R, T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(
    receiver: R, \n    completion: Continuation<T>\n): Any? =
    this.asDynamic() \n    (completion, false)\n
*\@SinceKotlin("1.3")\n@Suppress("UNCHECKED_CAST")\npublic fun <R, T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(
    receiver: R, \n    completion: Continuation<T>\n): Any? =
    this.asDynamic() \n    (completion, true)\n
this.asDynamic()(receiver, completion, false)/*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n*//*n
null
(midPoint < 0) return
    var reverseIndex = lastIndexOf
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*

/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun CharSequence.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.elementAt(index: Int): UInt {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun ULongArray.elementAt(index: Int): ULong {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.elementAt(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.elementAt(index: Int): UShort {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UIntArray.asList(): List<UInt> {
    return object : AbstractList<UInt>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UInt): Boolean = this@asList.contains(element)
        override fun get(index: Int): UInt {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UInt): Int {
            if ((element as Any?) !is UInt) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun ULongArray.asList(): List<ULong> {
    return object : AbstractList<ULong>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: ULong): Boolean = this@asList.contains(element)
        override fun get(index: Int): ULong {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: ULong): Int {
            if ((element as Any?) !is ULong) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
a [List] that wraps the original array.*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.js

import kotlin.annotation.AnnotationTarget.*

@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR,
        VALUE_PARAMETER, PROPERTY_GETTER, PROPERTY.Setter)
@Deprecated("Use `external` modifier instead", level = DeprecationLevel.ERROR)
public annotation class native(public val name: String = "")

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeGetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeSetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeInvoke

@Target(CLASS,
        PROPERTY)
@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.buttonset(): JQuery = asDynamic().buttonset()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(): JQuery = asDynamic().dialog()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(params: Json): JQuery = asDynamic().dialog(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(mode: String, param: String): Any? = asDynamic().dialog(mode, param)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(mode: String): JQuery = asDynamic().dialog(mode)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.button(): JQuery = asDynamic().button()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.accordion(): JQuery = asDynamic().accordion()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.draggable(params: Json): JQuery = asDynamic().draggable(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.selectable(): JQuery = asDynamic().selectable()
a class) specific name in JavaScript. This may be useful in the following cases:

There are two functions for which the compiler gives same name in JavaScript, you can mark one with `@JsName(...)` to prevent the compiler from reporting error.

You are writing a JavaScript library in Kotlin. The compiler produces mangled names for functions with parameters, which is unnatural for usual JavaScript developer.

You can put `@JsName(...)` on functions you want to be available from JavaScript.

For some reason you want to rename declaration, e.g. there's common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.

In case of file (which can't be `external`) the following rule applies: all the declarations in *the file must be `external`. By applying `@JsModule(…)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.

It's required to denote a valid JavaScript identifier.

In case of file (which can't be `external`) the following rule applies: all the declarations in *the file must be `external`. By applying `@JsModule(…)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.

It is not interpreted by the Kotlin compiler, it's passed as-is to the target module system.

In case of file (which can't be `external`) the following rule applies: all the declarations in *the file must be `external`. By applying `@JsModule(…)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.

It is not interpreted by the Kotlin compiler, it's passed as-is to the target module system.

In case of file (which can't be `external`) the following rule applies: all the declarations in *the file must be `external`. By applying `@JsModule(…)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.

It is not interpreted by the Kotlin compiler, it's passed as-is to the target module system.
or with the `-Xopt-in=kotlin.js.ExperimentalJsExport` compiler option is given.

```kotlin
/**
 * Exports top-level declaration.
 * @params E the type of elements contained in the collection. The collection is
 * invariant on its element type.
 */
@ExperimentalJsExport
@SinceKotlin("1.3")
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsExport
```

and
```
@Target(AnnotationTarget.FIELD)
@Retention(AnnotationRetention.SOURCE)
public actual annotation class Volatile
```

and
```
@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
public actual annotation class Synchronized
```
Open Source Used In DNAC Platform Arica 13307
AbstractMutableList<E>(), MutableList<E>, RandomAccess */

    /**
     * Creates an empty [ArrayList].

     * @param initialCapacity initial capacity (ignored)
     */
    public actual constructor() : this(emptyArray()) {}

    /**
     * Creates an empty [ArrayList].
     * @param initialCapacity initial capacity
     */
    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual constructor(initialCapacity: Int = 0) : this(emptyArray()) {}

    /**
     * Creates an [ArrayList] filled from the
     * elements collection.
     */
    public actual constructor(elements: Collection<E>) :
              this(elements.toTypedArray<Any?>()) {}

    /**\n     * Does nothing in this ArrayList implementation.
     */
    actual override fun trimToSize() {}

    /**\n     * Does nothing in this ArrayList implementation.
     */
    actual override fun ensureCapacity(minCapacity: Int) {}

    actual override val size: Int get() = array.size

    @Suppress("UNCHECKED_CAST")
    actual override fun get(index: Int): E = array[rangeCheck(index)] as E

    actual override fun set(index: Int, element: E): E {
        rangeCheck(index)
        return array[index].apply { array[index] = element } as E
    }

    actual override fun add(element: E): Boolean {
        array.asDynamic().push(element)
        modCount++
        return true
    }

    actual override fun add(index: Int, element: E): Unit {
        array.asDynamic().splice(insertionRangeCheck(index), 0, element)
        modCount++
    }

    actual override fun addAll(elements: Collection<E>): Boolean {
        if (elements.isEmpty()) return false
        array += elements.toTypedArray<Any?>()
        modCount++
        return true
    }

    actual override fun addAll(index: Int, elements: Collection<E>): Boolean {
        insertionRangeCheck(index)
        if (index == size) return addAll(elements)
        if (elements.isEmpty()) return false
        when (index) {
            size -> return addAll(elements)
            0 -> array = elements.toTypedArray<Any?>() + array
            else -> array = array.copyOfRange(0, index).asDynamic().concat(elements.toTypedArray<Any?>(), array.copyOfRange(index, size))
        }
        modCount++
        return true
    }

    actual override fun removeAt(index: Int): E {
        rangeCheck(index)
        modCount++
        return if (index == lastIndex)
            array.asDynamic().pop()
        else
            array.asDynamic().splice(index, 1)[0]
    }

    actual override fun remove(element: E): Boolean {
        for (index in array.indices) {
            if (array[index] == element) {
                array.asDynamic().splice(index, 1)
                modCount++
                return true
            }
        }
        return false
    }

    override fun removeRange(fromIndex: Int, toIndex: Int) {
        modCount++
        array.asDynamic().splice(fromIndex, toIndex - fromIndex)
    }

    actual override fun clear() {
        array = emptyArray()
        modCount++
    }

    actual override fun indexOf(element: E): Int = array.indexOf(element)

    actual override fun lastIndexOf(element: E): Int = array.lastIndexOf(element)

    override fun toString() = arrayToString(array)

    override fun toArray(): Array<Any?> = js("["]").slice.call(array)

    private fun rangeCheck(index: Int) = index.apply {
        AbstractList.checkElementIndex(index, size)
    }

    private fun insertionRangeCheck(index: Int) = index.apply {
        AbstractList.checkPositionIndex(index, size)
    }
}*/

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

internal fun <T> sortArrayWith(array: Array<out T>, comparison: (T, T) -> Int) {
    if (getStableSortingIsSupported()) {
        array.asDynamic().sort(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>()::sort, 0, array.lastIndex, Comparator(comparison))
    }
}

internal fun <T> sortArrayWith(array: Array<out T>, comparator: Comparator<in T>) {
    if (getStableSortingIsSupported()) {
        array.asDynamic().sort(comparator)
    } else {
        mergeSort(array.unsafeCast<Array<T>>()::sort, 0, array.lastIndex, comparator)
    }
}

internal fun <T : Comparable<T>> sortArray(array: Array<out T>, comparison: (T, T) -> Int) {
    if (getStableSortingIsSupported()) {
        val comparison = { a: T, b: T -> if (a <= b) -1 else 1 }
        array.asDynamic().sortBy(comparison)
    } else {
        mergeSort(array.unsafeCast<Array<T>>()::sortBy, 0, array.lastIndex, naturalOrder())
    }
}

private var _stableSortingIsSupported: Boolean? = null
private fun getStableSortingIsSupported(): Boolean {
    return true
}
array.asDynamic().push(index)
val comparison = { a: Int, b: Int -> (a and 3) - (b and 3) }
array.asDynamic().sort(comparison)
for (index in 1 until array.size)
val a = array[index - 1]
val b = array[index]
if (((a and 3) == (b and 3) && a >= b) return false)
_stableSortingIsSupported = true
return true
阵

private fun <T> mergeSort(array: Array<T>, start: Int, endInclusive: Int, comparator: Comparator<in T>): Array<T> {
    val buffer = arrayOfNulls<Any?>(array.size).unsafeCast<Array<T>>()
    val result = mergeSort(array, buffer, start, endInclusive, comparator)
    if (result !== array) {
        result.forEachIndexed { i, v -> array[i] = v }
    }
    return result
}

// Both start and end are inclusive indices.
    if (start == end) {
        return array
    }
    val median = (start + end) / 2
    val left = mergeSort(array, buffer, start, median, comparator)
    val right = mergeSort(array, buffer, median + 1, end, comparator)
    val target = if (left === buffer) array else buffer
    var leftIndex = start
    var rightIndex = median + 1
    for (i in start..end) {
        when {
            leftIndex <= median && rightIndex <= end -> {
                val leftValue = left[leftIndex]
                val rightValue = right[rightIndex]
                if (comparator.compare(leftValue, rightValue) <= 0) {
                    target[i] = leftValue
                    leftIndex++
                } else {
                    target[i] = rightValue
                    rightIndex++
                }
            }
            leftIndex <= median -> {
                target[i] = left[leftIndex]
                leftIndex++
            }
            else /* rightIndex <= end */ -> {
                target[i] = right[rightIndex]
                rightIndex++
                Unit // TODO: Fix KT-31506
            }
        }
    }
    return target
}
val size: Int get() = this@HashMap.size
     *
     * Internal implementation of the map: either string-based or hashcode-based.
     */
    private val internalMap: InternalMap<K, V> private val equality:
        EqualityComparator
        internal constructor(internalMap: InternalMap<K, V>): super(){
            this.internalMap = internalMap
            this.equality = internalMap.equality
        }
        * Constructs an empty [HashMap] instance.
        */
        actual constructor(): this(InternalHashCodeMap(EqualityComparator.HashCode))
        *
        * Constructs an empty [HashMap] instance.
        */
        * @param initialCapacity the initial capacity (ignored)
        * @param loadFactor the load factor (ignored)
        * @throws IllegalArgumentException if the initial capacity or load factor are negative
        */
        @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
        actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) {
            this(initialCapacity, loadFactor) {
                require(initialCapacity >= 0) { "Negative initial capacity: $initialCapacity" }
                require(loadFactor >= 0) { "Non-positive load factor: $loadFactor" }
            }
        }
        actual constructor(initialCapacity: Int) :
            this(initialCapacity, 0.0f)
        *
        * Constructs an instance of [HashMap] filled with the contents of the specified [original] map.
        */
        actual constructor(original: Map<out K, V>): this() {
            this.putAll(original)
        }
        actual override fun clear() {
            internalMap.clear()
            structureChanged(this)
        }
        actual override fun containsKey(key: K): Boolean =
            internalMap.containsKey(key)
        actual override fun containsValue(value: V): Boolean =
            internalMap.any {
                equality.equals(it.value, value)
            }
        private var _entries: MutableSet<MutableMap.MutableEntry<K, V>>? =
            null
        actual override val entries: MutableSet<MutableMap.MutableEntry<K, V>>
            get() {
                if (_entries == null) {
                    _entries = createEntrySet()
                }
                return _entries!!
            }
        protected open fun createEntrySet():
            MutableSet<MutableMap.MutableEntry<K, V>> = EntrySet()
        actual
        override operator fun get(key: K): V? = internalMap.get(key)
        actual override fun put(key: K, value: V): V? =
            internalMap.put(key, value)
        actual override fun remove(key: K): V? =
            internalMap.remove(key)
        actual override val size: Int get() = internalMap.size
        */
    }
}

/**
     * Constructs the specialized implementation of [HashMap] with [String] keys, which stores the keys as properties of an * JS object without hashing them.
     */
    public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
        return HashMap<String, V>(InternalStringMap(EqualityComparator.HashCode)).apply {
            putAll(pairs) }
    }


actual override val size: Int = get() = map.size

/**
 * Creates a new instance of the specialized implementation of [HashSet] with the specified [String] elements.
 * A simple wrapper around JavaScriptObject to provide [java.util.Map]-like semantics for any
 * key type.
 * Implementation notes:
 * A key's value is the index in backingMap which should contain that key. Since several keys may
 * have the same hash, each value in hashCodeMap is actually an array containing all entries
 * whose keys share the same hash.
 */
internal class InternalHashCodeMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {

    private var backingMap: dynamic = createJsMap()
    override var size: Int = 0
        private set

    override fun put(key: K, value: V): V? {
        val hashCode = equality.getHashCode(key)
        val chainOrEntry = getChainOrEntryOrNull(hashCode) ?: return null
        // This is a new chain, put it to the map.
        val backingMap[hashCode] = SimpleEntry(key, value)
        if (chainOrEntry !is Array<*>) {
            // It is an entry
            val entry: SimpleEntry<K, V> = chainOrEntry
            if (equality.equals(entry.key, key)) {
                return entry.setValue(value)
            } else {
                backingMap[hashCode] = arrayOf(entry, SimpleEntry(key, value))
                size++
                return null
            }
        } else {
            // Chain already exists, perhaps key also exists.
            val chain = Array<MutableEntry<K, V>>()
            if (equality.equals(chainOrEntry, key)) {
                return null
            } else {
                val chain = chainOrEntry
                for (index in chain.indices) {
                    val entry = chain[index]
                    if (equality.equals(entry.key, key)) {
                        if (chain.size == 1) {
                            chain.asDynamic().length = 0
                            // remove the whole array
                            jsDeleteProperty(backingMap, hashCode)
                        } else {
                            // splice out the entry we're removing
                            chain.asDynamic().splice(index, 1)
                            size--
                            structureChanged(host)
                            return null
                        }
                    }
                }
            }
            return null
        }

    override fun remove(key: K): V? {
        val hashCode = equality.getHashCode(key)
        val chainOrEntry = getChainOrEntryOrNull(hashCode) ?: return null
        if (chainOrEntry !is Array<*>) {
            return null
        } else {
            val chain = chainOrEntry
            return null
        }
    }

    override fun clear() {
        backingMap = createJsMap()
        size = 0
    }

    override fun contains(key: K): Boolean = getEntry(key) != null

    override fun get(key: K): V? = getEntry(key)?.value

    private fun getEntry(key: K): MutableEntry<K, V>? {
        val chainOrEntry = getChainOrEntryOrNull(equality.getHashCode(key)) ?: return null
        if (chainOrEntry !is Array<*>) {
            val entry = chainOrEntry
            if (equality.equals(entry.key, key)) {
                return entry
            } else {
                return null
            }
        } else {
            val chain = chainOrEntry
            return chain.findEntryInChain(key)
        }

    private fun Array<MutableEntry<K, V>>.findEntryInChain(key: K): MutableEntry<K, V>? =
        firstOrNull { entry ->
            equality.equals(entry.key, key)
        }

    override fun iterator(): MutableIterator<MutableEntry<K, V>> {

        return object : MutableIterator<MutableEntry<K, V>> {
            var state = -1 // -1 not ready, 0 - ready, 1 - done
            var chainOrEntry: dynamic = null
            var isChain = false
            var itemIndex = -1
            var lastEntry: MutableEntry<K, V>? = null
            override fun hasNext(): Boolean {
                return true
            }

            override fun next(): MutableEntry<K, V> {

                if (chainOrEntry != null &&
                    var chainOrEntry: dynamic = null
                    var isChain = false
                    var itemIndex = -1
                    var lastEntry: MutableEntry<K, V>? = null
                    private fun computeNext(): Int {
                        if (chainOrEntry != null &&
                            var chainOrEntry: dynamic = null
                            var isChain = false
                            var itemIndex = -1
                            var lastEntry:.MutableEntry<K, V>? = null
                            private fun hasNext(): Boolean {
                                return true
                            }
                        } else {
                            return null
                        }
                    }
                    override fun next(): MutableEntry<K, V> {

                        if (chainOrEntry != null &&
                        var chainOrEntry: dynamic = null
                        var isChain = false
                        var itemIndex = -1
                        var lastEntry:MutableEntry<K, V>? = null
                        private fun computeNext(): Int {
                            if (chainOrEntry != null &&
                                var chainOrEntry: dynamic = null
                                var isChain = false
                                var itemIndex = -1
                                var lastEntry:MutableEntry<K, V>? = null
                                private fun hasNext(): Boolean {
                                    return true
                                }
                            } else {
                                return null
                            }
                        }
                    }
                }
            }
if (++itemIndex < chainSize)
                        return 0
                    } else {
                        chainOrEntry = null
                    }
                    itemIndex = 0
                    return 0
                }

                if (++keyIndex < keys.size)
                    chainOrEntry = backingMap[keys[keyIndex]]
                    isChain = chainOrEntry is Array<*>
                    itemIndex = 0
                    return 0
                } else {
                    chainOrEntry = null
                    return 1
                }

            override fun hasNext(): Boolean {
                if (state == -1)
                    state = computeNext()
                return state == 0
            }

            override fun next(): MutableEntry<K, V> {
                if (!hasNext()) throw NoSuchElementException()
                val lastEntry = if (isChain) {
                    chainOrEntry.unsafeCast<Array<MutableEntry<K, V>>>()[itemIndex]
                } else {
                    chainOrEntry.unsafeCast<MutableEntry<K, V>>()
                }
                this.lastEntry = lastEntry
                state = -1
                return lastEntry
            }

            override fun remove() {
                checkNotNull(lastEntry)
                this@InternalHashCodeMap.remove(lastEntry!!.key)
                lastEntry = null
                // the chain being iterated just got modified by InternalHashCodeMap.remove
                itemIndex--
            }

            private fun getChainOrEntryOrNull(hashCode: Int): dynamic {
                val chainOrEntry = backingMap[hashCode]
                return if (chainOrEntry === undefined) null else chainOrEntry
            }
override fun remove() {  
    @Suppress("UNCHECKED_CAST")
    return newMapEntry(key as K)
}

override fun remove() {  
    @Suppress("UNCHECKED_CAST")
    this@InternalStringMap.remove(checkNotNull(lastKey) as K)
}

private fun newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {  
    override val key: K get() = key
    override val value: V get() = this@InternalStringMap[get()]

    override fun setValue(newValue: V): V = this@InternalStringMap.put(key, newValue)

    override fun hashCode(): Int = AbstractMap.entryHashCode(this)
    override fun toString(): String = AbstractMap.entryToString(this)
    override fun equals(other: Any?): Boolean = AbstractMap.entryEquals(this, other)
}
ChainEntry<K, V>.remove() { if (this.next == this) { // if this is single element, remove head
        head = null
    } else { if (head == this) { // if this is first element, move head to next
        head = next
    } else { next!!.prev = prev
            prev!!.next = next
        next = null
        prev = null
    } }
}

/*
 * The hashmap that keeps track of our entries and the chain. Note that we
 * duplicate the key here to eliminate changes to HashMap and minimize the
 * code here, at the expense of additional space.
 */
private val map: HashMap<K, ChainEntry<K, V>>

/*
 * Constructs an empty [LinkedHashMap] instance.
 */
actual constructor() : super() {
    map = HashMap<K, ChainEntry<K, V>>()
}

/*
 * Constructs an empty [LinkedHashMap] instance.
 */
constructor(backingMap: HashMap<K, Any>) : this(0)

/*
 * Constructs an empty [LinkedHashMap] instance.
 */
constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(initialCapacity, loadFactor)

/*
 * Constructs an empty [LinkedHashMap] instance.
 */
constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

/*
 * Constructs a new [LinkedHashMap] instance filled with the contents of the specified [original] map.
 */
constructor(original: Map<out K, V>) : this(0)

/*
 * Constructs a new [LinkedHashSet] instance.
 */
actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(initialCapacity, loadFactor)

/*
 * Constructs a new [LinkedHashSet] instance.
 */
constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

actual constructor(original: Collection<K, V>) : this(0)

actual override fun clear() {
    map.clear()
    head = null
}

actual override fun get(key: K): V? = map.get(key)?.value

actual override fun put(key: K, value: V): V? {
    val old = map.get(key)
    if (old == null) {
        val newEntry = ChainEntry(key, value)
        map.put(key, newEntry)
        newEntry.addToEnd()
        return null
    } else {
        return old.setValue(value)
    }
}

actual override fun remove(key: K): V? {
    val entry = map.remove(key)
    if (entry != null) {
        entry.remove()
        return entry.value
    }
    return null
}

actual override fun size: Int get() = map.size

actual override fun values(): Collection<V> = map.values().toMutableSet()
of JS object without hashing them.

*/
public fun linkedStringSetOf(vararg elements: String):
    LinkedHashSet<String> {
        return LinkedHashSet(linkedStringMapOf<Any>()).apply { addAll(elements)
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.contracts.*


@Deprecated("Do not use Synchronized annotation in pure Kotlin/JS code", level = DeprecationLevel.ERROR)
public typealias Synchronized = kotlin.jvm.Synchronized

@Deprecated("Do not use Volatile annotation in pure Kotlin/JS code", level = DeprecationLevel.ERROR)
public typealias Volatile = kotlin.jvm.Volatile

@kotlin.internal.InlineOnly
public
    actual inline fun <R> synchronized(lock: Any, block: () -> R): R {
        contract {
            callsInPlace(block, InvocationKind.EXACTLY_ONCE)
        }
        return block()
    }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.io

internal abstract class BaseOutput {
    open fun println() {
        print("\n")
    }

    open fun println(message: Any?) {
        print(message)
        println()
    }

    abstract fun print(message: Any?)

    open fun flush() {}
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("NodeJsOutput")
internal class NodeJsOutput(val outputStream: dynamic) : BaseOutput() {
    override fun print(message: Any?) {
        // TODO: Using local variable because of bug in block decomposition lowering in IR backend
        val messageString = String(message)
        outputStream.write(messageString)
    }

    override fun println(message: Any?) {
        console.log(message)
    }

    override fun println() {
        console.log("")
    }
}

/** JsName used to make the declaration available outside of module to test it and use at try.kotl.in */
@JsName("BufferedOutput")
internal open class BufferedOutput : BaseOutput() {
    var buffer = ""

    override fun print(message: Any?) {
        buffer += String(message)
    }

    override fun flush() {
        buffer = ""
    }
}

/** JsName used to make the declaration available outside of module to test it */
@JsName("BufferedOutputToConsoleLog")
internal class BufferedOutputToConsoleLog : BufferedOutput() {
    override fun print(message: Any?) {
        var s = String(message)
        val i = s.nativeLastIndexOf("\n", 0)
        if (i >= 0) {
            buffer += s.substring(0, i)
            flush()
            s = s.substring(i + 1)
        }
        buffer += s
    }
}

/** Prints the line separator to the standard output stream. */
public actual fun println() {
    output.println()
}

/** Prints the given [message] and the line separator to the standard output stream. */
public actual fun println(message: Any?) {
    output.println(message)
}

/** Prints the given [message] to the standard output stream. */
public actual fun print(message: Any?) {
    output.print(message)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines

import kotlin.coroutines.intrinsics.CoroutineSingletons.*
import kotlin.coroutines.intrinsics.COROUTINE_SUSPENDED

@PublishedApi
@SinceKotlin("1.3")
internal
    actual class SafeContinuation<in T> : Continuation<T> {

    @PublishedApi
    internal actual constructor(delegate: Continuation<T>) : this(delegate, UNDECIDED)

    /**
     * @param delegate the delegate Continuation
     * @param initialResult the initial result
     */
    actual constructor(delegate: Continuation<T>, initialResult: Any?) : this(delegate, initialResult)

    actual override fun resumeWith(result: Result<T>?) {
        when (this.result) {
            UNDECIDED -> {
                this.result = result.value
            } else -> throw
IllegalStateException("Already resumed")
        }
    }

    @PublishedApi
    internal actual fun getOrThrow(): Any? {
        if (result === UNDECIDED) {
            result = COROUTINE_SUSPENDED
            return COROUTINE_SUSPENDED
        }
        val result = this.result
        return when {
            result === RESUMED -> COROUTINE_SUSPENDED // already called continuation, indicate COROUTINE_SUSPENDED upstream
            result is Result.Failure -> throw result.exception
            else -> result // either COROUTINE_SUSPENDED or data
        }
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.coroutines.js.internal

import kotlin.coroutines.Continuation
import kotlin.coroutines.EmptyCoroutineContext

@PublishedApi
@SinceKotlin("1.3")
internal val EmptyContinuation = Continuation<Any?>(EmptyCoroutineContext) { result ->
    result.getOrThrow()
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.js

/**
 * Exposes the [Date API](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date) to Kotlin.
 * @Suppress("NOT_DOCUMENTED")
 */
public external class Date() {
    public constructor(milliseconds: Number)
    public constructor(dateString: String)
    public constructor(year: Int, month: Int)
    public constructor(year: Int, month: Int, day: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int)
    public constructor(year: Int, month: Int, day: Int, hour: Int, minute: Int, second: Int, millisecond: Number)
    public fun getDate(): Int
    public fun getDay(): Int
    public fun getFullYear(): Int
    public fun getHours(): Int
    public fun getMilliseconds(): Int
    public fun getMinutes(): Int
    public fun getMonth(): Int
    public fun getSeconds(): Int
    public fun getTime(): Double
    public fun getTimezoneOffset(): Int
    public fun getUTCDate(): Double
    public fun getUTCDay(): Int
    public fun getUTCFullYear(): Int
    public fun getUTCHours(): Int
    public fun getUTCMilliseconds(): Int
    public fun getUTCMinutes(): Int
    public fun getUTCMonth(): Int
    public fun getUTCSeconds(): Int
    public fun toDateString(): String
    public fun toISOString(): String
    public fun toJSON(): Json
    public fun toLocaleDateString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String
    public fun toLocaleTimeString(locales: String, options: LocaleOptions = definedExternally): String
    public fun toTimeString(): String
    public fun toUTCString(): String
    public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
        val result = js("new Object()").unsafeCast<Date.LocaleOptions>()
        init(result)
        return result
    }
}

public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
    val result = js("new Object()").unsafeCast<Date.LocaleOptions>()
    init(result)
    return result
}
Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.dom

import org.w3c.dom.*

/**
 * Creates a new element with the specified [name].
 * The element is initialized with the specified [init] function.
 * This is a public function.
 */
public fun Document.createElement(name: String, init: Element.() -> Unit): Element = createElement(name).apply(init)

/**
 * Appends a newly created element with the specified [name]
 * to this element.
 * The element is initialized with the specified [init] function.
 */
public fun Element.appendElement(name: String, init: Element.() -> Unit): Element =

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.dom

import org.w3c.dom.*

fun Element.hasClass(cssClass: String): Boolean =
    className.matches("\b\$cssClass\$\")

fun Element.addClass(vararg cssClasses: String): Boolean {
    val missingClasses = cssClasses.filterNot { hasClass(it) }
    if (missingClasses.isNotEmpty()) {
        val presentClasses = className.trim()
        className = buildString {
            append(presentClasses)
            if (presentClasses.isEmpty()) {
                append(" ")
            }
            missingClasses.joinTo(this, " ")
        }()
        return true
    }

    return false
}

fun Element.removeClass(vararg cssClasses: String): Boolean {
    if (cssClasses.any { hasClass(it) }) {
        val toBeRemoved = cssClasses.toSet()
        className = className.trim().split("\\s+".toRegex()).filter { it !in toBeRemoved }.joinToString(" ")
        return true
    }

    return false
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("StringsKt")

package kotlin.text

/**
 * A mutable sequence of characters.
 * A string builder can be used to efficiently perform multiple string manipulation operations.
 * Constructs a string builder with the specified initial [capacity].
 * Constructs a string builder that contains the same characters as the specified [content] string.
 * @Since Kotlin("1.3")
 * @ExperimentalStdlibApi
 */
constructor(content: String)
 constructor(capacity: Int)
 constructor(content: CharSequence)

/**
 * Converts the string into a regular expression [Regex] with the default options.
 * Converts the string into a regular expression [Regex] with the specified single [option].
 * Converts the string into a regular expression [Regex] with the specified set of [options].
 * Converts the string into a regular expression [Regex] with the specified initial [capacity].
 * Constructs a string builder that contains the same characters as the specified [content] string.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
constructor(content: String)
```
were unpaired low-surrogates and high-surrogates before the operation.

* For example, reversing `"\uDC00\uDC00"` produces `"\uDC00\uDC00"` which is a valid surrogate pair.

```kotlin
fun reverse(): StringBuilder
```
same order as in the [value] character sequence, starting at [index].
* @param index the position in this string builder to insert at.
* @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `"null"` are inserted.
* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun insert(index: Int, value: CharSequence?): StringBuilder

/**
 * Inserts the string [value] into this string builder at the specified [index] and returns this instance.
* @param index the position in this string builder to insert at.
* @param value the string to insert at this position.
* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun insert(index: Int, value: String): StringBuilder

/**
 * Inserts the string [value] into this string builder at the specified [index] and returns this instance.
* @param index the position in this string builder to insert at.
* @param value the string to insert at this position.
* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun insert(index: Int, value: Any?): StringBuilder

/**
 * Inserts the character sequence [value] into this string builder at the specified [index] and returns this instance.
* @param index the position in this string builder to insert at.
* @param value the character sequence from which characters are inserted. If [value] is `null`, then the four characters `"null"` are inserted.
* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun insert(index: Int, value: CharSequence?): StringBuilder

/**
 * Sets the length of this string builder to the specified [newLength] and returns this instance.
* @param newLength the new length.
* @throws IndexOutOfBoundsException if [newLength] is less than zero.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun setLength(newLength: Int)

/**
 * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).
* @param startIndex the beginning (inclusive) of the range to replace.
* @param endIndex the end (exclusive) of the range to replace.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun substring(startIndex: Int, endIndex: Int): String

/**
 * Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [length] (exclusive).
* @param startIndex the beginning (inclusive) of the range to replace.
* @param endIndex the end (exclusive) of the range to replace.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
fun substring(startIndex: Int): String

/**
 * Removes the character at the specified [index] and returns this instance.
* @param index the position in this string builder to remove.
* @throws IndexOutOfBoundsException if [index] is less than zero, greater than the length of this string builder, or `startIndex > endIndex`.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun StringBuilder.deleteAt(index: Int): StringBuilder

/**
 * Removes characters in the specified range from this string builder and returns this instance.
* @param startIndex the beginning (inclusive) of the range to remove.
* @param endIndex the end (exclusive) of the range to remove.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder

/**
 * Removes characters in the specified range from this string builder and returns this instance.
* @param startIndex the beginning (inclusive) of the range to remove.
* @param endIndex the end (exclusive) of the range to remove.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
* @SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun StringBuilder.trimToSize()
instance.

```kotlin
@param startIndex the beginning (inclusive) of the range to remove.
@throws IndexOutOfBoundsException or [IllegalStateException] when [startIndex] is out of range of the string builder indices or when `startIndex > endIndex`.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of the [value] array or when `startIndex > endIndex`.

@throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of the [value] array.
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public expect fun StringBuilder.insertRange(index: Int, value: CharSequence?, startIndex: Int, endIndex: Int):
```
StringBuilder

@Suppress("EXTENSION_SHADOWED_BY_MEMBER")
@Deprecated("Use append(value: Any?) instead", ReplaceWith("append(value = obj)"),
DeprecationLevel.WARN)
public inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)

/**
 * Builds new string by populating newly created [StringBuilder] using
 * provided [builderAction] and then converting it to [String].
 */
public fun StringBuilder.appendBuilderAction(builderAction: () -> Unit): String =
StringBuilder().apply(builderAction).toString()

/**
 * Append all arguments to the given [StringBuilder].
 */
public fun <T> StringBuilder.append(vararg value: T): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/**
 * Appends all arguments to the given [StringBuilder].
 */
public fun StringBuilder.append(vararg value: Any?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

/package.kotlin

import org.w3c.dom.*

/**
 * Gets a value indicating whether this node is a TEXT_NODE or a CDATA_SECTION_NODE.
 */
public val Node.isText: Boolean
    get() = nodeType == Node.TEXT_NODE || nodeType == Node.CDATA_SECTION_NODE

/**
 * Gets a value indicating whether this node is an [Element].
 */
public val Node.isElement: Boolean
    get() = nodeType == Node.ELEMENT_NODE

/package.kotlin.js

/**
 * Reinterprets this value as a value of the [dynamic type](/docs/reference/dynamic-type.html).
 */
@kotlin.internal.InlineOnly
public inline fun Any?.asDynamic(): dynamic = this

/**
 * Reinterprets this value as a value of the specified type [T] without any actual type checking.
 */
@kotlin.internal.NoInfer
@kotlin.internal.InlineOnly
public inline fun <T> Any?.unsafeCast(): @kotlin.internal.NoInfer T = this

/package.kotlin.js@Modifier"unsafeCastDynamic()"@kotlin.internal.InlineOnly

object in the following cases:
* when it has an `iterator` function,
* when it is an array
* when it is an instance of [kotlin.collections.Iterable]
* `@kotlin.internal.DynamicExtension`\n\nfunc dynamic.iterator(): Iterator<dynamic> {
    val r: Any? = this

    return when {
        this["iterator"] != null ->
            this["iterator"]()

        isArrayish(r) ->
            r.unsafeCast<Array<*>>().iterator()\n
        else ->

            (r as Iterable<*>).iterator()\n    }
}

package kotlin

// NOTE: Do not author your exceptions as they are written in this file, instead use this template:

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

public open class MyException : Exception {
    constructor() : super()
    constructor(message: String?) : super(message)
    constructor(message: String?, cause: Throwable?) : super(message, cause)
    constructor(cause: Throwable?) : super(cause)
}

public actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor() : this(null, null) {
        Error::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        Error::class.js.asDynamic().call(this, message, null)
    }

    actual constructor(cause: Throwable?) : this(undefine, cause) {
        Error::class.js.asDynamic().call(this, undefined, cause)
    }

    actual constructor(message: String?, cause: Throwable?) : this(message, cause) {
        Error::class.js.asDynamic().call(this, message, cause)
    }
}

public actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor() : this(null, null) {
        Exception::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        Exception::class.js.asDynamic().call(this, message, null)
    }

    actual constructor(cause: Throwable?) : this(undefine, cause) {
        Exception::class.js.asDynamic().call(this, undefined, cause)
    }
}

public actual open class RuntimeException actual constructor(message: String?, cause: Throwable?) : Exception(message, cause) {
    actual constructor() : this(null, null) {
        RuntimeException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        RuntimeException::class.js.asDynamic().call(this, message, null)
    }

    actual constructor(cause: Throwable?) : this(undefine, cause) {
        RuntimeException::class.js.asDynamic().call(this, undefined, cause)
    }
}

public actual open class IllegalArgumentException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null) {
        IllegalArgumentException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        IllegalArgumentException::class.js.asDynamic().call(this, message, null)
    }
}

public actual open class IllegalStateException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null) {
        IllegalStateException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        IllegalStateException::class.js.asDynamic().call(this, message, null)
    }
}

public actual open class IndexOutOfBoundsException actual constructor(message: String?) : RuntimeException(message) {
    actual constructor() : this(null, null) {
        IndexOutOfBoundsException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        IndexOutOfBoundsException::class.js.asDynamic().call(this, message, null)
    }
}

public actual open class ConcurrentModificationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null) {
        ConcurrentModificationException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        ConcurrentModificationException::class.js.asDynamic().call(this, message, null)
    }

    actual constructor(cause: Throwable?) : this(undefine, cause) {
        ConcurrentModificationException::class.js.asDynamic().call(this, undefined, cause)
    }
}

public actual open class UnsupportedOperationException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null) {
        UnsupportedOperationException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        UnsupportedOperationException::class.js.asDynamic().call(this, message, null)
    }

    actual constructor(cause: Throwable?) : this(undefine, cause) {
        UnsupportedOperationException::class.js.asDynamic().call(this, undefined, cause)
    }
}

public actual open class NumberFormatException actual constructor(message: String?) : IllegalArgumentException(message) {
    actual constructor() : this(null, null) {
        NumberFormatException::class.js.asDynamic().call(this, null, null)
    }

    actual constructor(message: String?) : this(message, null) {
        NumberFormatException::class.js.asDynamic().call(this, message, null)
    }
}

Open Source Used In DNAC Platform Arica  13322
NullPointerException actual constructor(message: String?) : RuntimeException(message) {
    actual constructor() : this(null)
}

public actual open class ClassCastException actual constructor(message: String?) : RuntimeException(message) {
    actual constructor() : this(null)
}

public actual open class AssertionError
private constructor(message: String?, cause: Throwable?) : Error(message, cause) {
    actual constructor() : this(null)
    constructor(message: String?) : this(message, null)
    actual constructor(message: Any?) : this(message.toString(), message as? Throwable)
}

public actual open class NoSuchElementException actual constructor(message: String?) : RuntimeException(message) {
    actual constructor() : this(null)
}

@SinceKotlin("1.3")
public actual open class ArithmeticException actual constructor(message: String?) : RuntimeException(message) {
    actual constructor() : this(null)
}

public actual open class NoWhenBranchMatchedException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

public actual open class UninitializedPropertyAccessException actual constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause) {
    actual constructor() : this(null, null)
    actual constructor(message: String?) : this(message, null)
    actual constructor(cause: Throwable?) : this(cause)
}

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

kotlin.collections

/**
 * Groups elements from the [Grouping] source by key and counts elements in each
 * group.
 */
@SinceKotlin("1.1")
public actual fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) {
        acc, _ -> acc + 1
    }

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.collections

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each
 * element to get its key. `keyOf` is a function that can be applied to each
 * element to get its key.
 */
@SinceKotlin("1.1")
public interface Grouping<T, out K> {
    /** Returns an [Iterator] over the elements of the source of this grouping. */
    fun sourceIterator(): Iterator<T>
    /** Extracts the key of an [element]. */
    fun keyOf(element: T): K
}

/**
 * Groups elements from the
 * [Grouping] source by key and applies [operation] to the elements of each group
 * sequentially, passing the previously accumulated value and the current element as arguments, and stores
 * the results in a new map.
 */
@SinceKotlin("1.1")
    return aggregateTo(operand)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

Open Source Used In DNAC Platform Arica 13323
previously accumulated value and the current element as arguments, and stores the results in the given destination map. The key for each element is provided by the [Grouping.keyOf] function. A parameter operation is a function that is invoked on each element with the following parameters: the key of the group this element belongs to, the current value of the accumulator of the group, and the element from the source being accumulated. If the first element encountered in the group is the first element encountered in the group, then the elements being aggregated for that key are never considered. The function returns a [Map] associating the key of each group with the result of aggregation of the group's elements. The function returns a [Map] associating the key of each group with the result of aggregation of the group's elements.

```kotlin
@SinceKotlin("1.1")
public inline fun <T, K, R : MutableMap<in K, R>> Grouping<T, K>.aggregateTo(
    destination: M,
    operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R
): M =
    @Suppress("UNCHECKED_CAST")
    aggregate { key, acc, e, first -> operation(key, if (first) accumulator else acc as R, e) }

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in a new map. An initial value of accumulator is provided by [defaultValueSelector] function. It's invoked with parameters: the key of the group, the first element being encountered in that group, and the following parameters: the key of the group this element belongs to, the current value of the accumulator of the group, and the element from the source being accumulated. The function returns a [Map] associating the key of each group with the result of aggregating the group's elements.

```
result of accumulating the group elements.

* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue

```kotlin
@public inline fun <T, K, R : @SinceKotlin("v1.1.1")public inline fun <T, K, R> Grouping<T, K>.foldByEvenLengthWithConstantInitialValue(R): R operation: (accumulator: R, element: T) -> R

@Suppress("UNCHECKED_CAST")
Map<K, R> =
    @Suppress("UNCHECKED_CAST")
    aggregateTo(destination) { _, acc, e, first -> operation(if (first)

```

```kotlin
* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue
```

```kotlin
* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue

```

```kotlin
* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue
```

```kotlin
* @sample samples.collections.Grouping.foldByEvenLengthWithConstantInitialValue
```
corresponding to the key of some group.

*An* return the [destination] map associating the key of each group with the sum of elements in the group.

```kotlin
@SinceKotlin("1.1")
    eachSumOfTo(destination: M, valueSelector: (T) -> Int): M =
        foldTo(destination, 0) { acc, e -> acc + valueSelector(e)}
    // TODO: sum by long and by double overloaded
    public fun <T, K : MutableList<K>> Grouping<T, K>.
    sumEachByLongTo(destination: M, valueSelector: (T) -> Long): M =
        foldTo(destination, 0L) { acc, e -> acc + valueSelector(e)}
    public fun <T, K> Grouping<T, K>.
    sumEachByDoubleTo(destination: M, valueSelector: (T) -> Double): M =
        foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e)}
```

---

@package kotlin

```kotlin
/**
 */
@Deprecated("Use top-level functions from kotlin.math package instead.", level = DeprecationLevel.WARNING)
public object Math {
    @Deprecated("Use kotlin.math.PI instead.", level = DeprecationLevel.ERROR)
    public val PI: Double
    @Deprecated("Use Random.nextDouble instead", ReplaceWith("kotlin.random.Random.nextDouble()", "kotlin.random")
                level = DeprecationLevel.ERROR)
    public fun random(): Double
    @Deprecated("Use kotlin.math.abs instead.", ReplaceWith("kotlin.math.abs(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun abs(value: Double): Double
    @Deprecated("Use kotlin.math.acos instead.", ReplaceWith("kotlin.math.acos(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun acos(value: Double): Double
    @Deprecated("Use kotlin.math.asin instead.", ReplaceWith("kotlin.math.asin(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun asin(value: Double): Double
    @Deprecated("Use kotlin.math.atan instead.", ReplaceWith("kotlin.math.atan(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun atan(value: Double): Double
    @Deprecated("Use kotlin.math.atan2 instead.", ReplaceWith("kotlin.math.atan2(y, x)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun atan2(y: Double, x: Double): Double
    @Deprecated("Use kotlin.math.cos instead.", ReplaceWith("kotlin.math.cos(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun cos(value: Double): Double
    @Deprecated("Use kotlin.math.sin instead.", ReplaceWith("kotlin.math.sin(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun sin(value: Double): Double
    @Deprecated("Use kotlin.math.exp instead.", ReplaceWith("kotlin.math.exp(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun exp(value: Double): Double
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Int): Int
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Float): Float
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Double): Double
    @Deprecated("Use kotlin.math.cos instead.", ReplaceWith("kotlin.math.cos(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun cos(value: Double): Double
    @Deprecated("Use kotlin.math.sin instead.", ReplaceWith("kotlin.math.sin(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun sin(value: Double): Double
    @Deprecated("Use kotlin.math.exp instead.", ReplaceWith("kotlin.math.exp(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun exp(value: Double): Double
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Int): Int
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Float): Float
    @Deprecated("Use maxOf or kotlin.math.max instead", level = DeprecationLevel.ERROR)
    public fun max(vararg values: Double): Double
    @Deprecated("Use minOf or kotlin.math.min instead", level = DeprecationLevel.ERROR)
    public fun min(vararg values: Int): Int
    @Deprecated("Use minOf or kotlin.math.min instead", level = DeprecationLevel.ERROR)
    public fun min(vararg values: Float): Float
    @Deprecated("Use minOf or kotlin.math.min instead", level = DeprecationLevel.ERROR)
    public fun min(vararg values: Double): Double
    @Deprecated("Use kotlin.math.sqrt instead.", ReplaceWith("kotlin.math.sqrt(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun sqrt(value: Double): Double
    @Deprecated("Use kotlin.math.tan instead.", ReplaceWith("kotlin.math.tan(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun tan(value: Double): Double
    @Deprecated("Use kotlin.math.ln instead.", ReplaceWith("kotlin.math.ln(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun log(value: Double): Double
    @Deprecated("Use kotlin.math.pow instead.", ReplaceWith("kotlin.math.pow(value)", "kotlin.math")
                level = DeprecationLevel.ERROR)
    public fun pow(value: Double): Double
```
pow(base: Double, exp: Double): Double
@Deprecated("Use kotlin.math.pow instead.",
ReplaceWith("kotlin.pow(base, exp)", "kotlin.math.pow(base, exp)"), level = DeprecationLevel.ERROR)
public fun pow(value: Number, base: Double, exp: Double): Double

Int
@Deprecated("Use kotlin.math.round instead.",
ReplaceWith("round(value)", "kotlin.math.round"), level = DeprecationLevel.ERROR)
public fun round(value: Number): Int

Int
@Deprecated("Use kotlin.math.floor instead.",
ReplaceWith("floor(value)", "kotlin.math.floor"), level = DeprecationLevel.ERROR)
public fun floor(value: Number): Int

Int
@Deprecated("Use kotlin.math.ceil instead.",
ReplaceWith("ceil(value)", "kotlin.math.ceil"), level = DeprecationLevel.ERROR)
public fun ceil(value: Number): Int

@PublishedApi
internal fun trunc(value: Number): Double
@PublishedApi
internal fun sign(value: Number): Double

@PublishedApi
internal fun sinh(value: Double): Double
@PublishedApi
internal fun cosh(value: Double): Double
@PublishedApi
internal fun tanh(value: Double): Double
@PublishedApi
internal fun asinh(value: Double): Double
@PublishedApi
internal fun acosh(value: Double): Double

@PublishedApi
internal fun atanh(value: Double): Double

@PublishedApi
internal fun hypot(x: Double, y: Double): Double

@PublishedApi
internal fun expm1(value: Double): Double

@PublishedApi
internal fun log10(value: Double): Double
@PublishedApi
internal fun log2(value: Double): Double
@PublishedApi
internal fun log1p(value: Double): Double

@PublishedApi
internal fun clz32(value: Int): Int

/**
 * Returns the smaller of two values.
 */
@Suppress("DEPRECATION")
@Deprecated("Use minOf or kotlin.math.min instead",
ReplaceWith("minOf(a, b)", "kotlin.math.min(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.min(a: Long, b: Long): Long =
    if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
@Suppress("DEPRECATION")
@Deprecated("Use maxOf or kotlin.math.max instead",
ReplaceWith("maxOf(a, b)", "kotlin.math.max(a, b)"), level = DeprecationLevel.ERROR)
public fun Math.max(a: Long, b: Long): Long =
    if (a >= b) a else b

/**
 * An interface for indexing access to a collection of
 * key-value pairs, where type of key is [String] and type of value is [Any?][Any].
 */
public external interface Json
{
    /**
     * Calls to the function will be translated to indexing operation (square brackets) on the receiver with
     * [propertyName] as the argument.
     * E.g. for next code:
     * ```kotlin
     * fun test(j: Json, p: String) = j["prop"] + j.get(p)
     * ```
     * will be generated:
     * ```js
     * function test(j, p) {
     *     return j["prop"] + j[p];
     * }
     * ```
     */
    operator fun get(propertyName: String): Any?

    /**
     * Calls
     * of the function will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].
     * E.g. for the following code:
     * ```kotlin
     * fun test(j: Json, p: String, newValue: Any) {
     *     j["prop"] = 1
     *     j.set(p, newValue)
     * }
     * ```
     * will be generated:
     * ```js
     * function test(j, p, newValue) {
     *     j["prop"] = 1;
     *     j[p] = newValue;
     * }
     * ```
     */
    operator fun set(propertyName: String, value: Any?): Unit
}

/**
 * Returns a
 * simple JavaScript object (as [Json]) using provided key-value pairs as names and values of its properties.
 */
public fun json(vararg pairs: Pair<String, Any?>): Json {
    val res: dynamic = js("({})")
    for ((name, value) in pairs) {
        res[name] = value
    }
    return res
}

/**
 * Adds key-value pairs from [other] to
 * [this].
 * Returns the original receiver.
 */
public fun Json.add(other: Json): Json {
    val keys: Array<String> = js("Object").keys(other)
    for (key in keys) {
        if (other.asDynamic().hasOwnProperty(key)) {
            this[key] = other[key];
        }
    }
    return this
}

/**
 */
@Suppress("NOT_DOCUMENTED")
public external object JSON {
    public fun stringify(o: Any?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?: String
    public fun stringify(o: Any?, replacer: Array<String>): String
    public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
    public fun loadString(string: String): Json
}

Open Source Used In DNAC Platform Arica  13327
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/
@file:Suppress("DEPRECATION", "DEPRECATION_ERROR")
package kotlin.math

import kotlin.internal.InlineOnly
import kotlin.js.Math as nativeMath

// region

// Double Math
================ Double Math ========================================

/** Computes the sine of the angle [x] given in radians. *
 * Special cases:
 * - `sin(NaN|+Inf|-Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun sin(x: Double): Double = nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians. *
 * Special cases:
 * - `cos(NaN|+Inf|-Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun cos(x: Double): Double = nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians. *
 * Special cases:
 * - `tan(NaN|+Inf|-Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun tan(x: Double): Double = nativeMath.tan(x)

/** Computes the arc sine of the value [x]; the returned value is an angle in the range from -PI/2 to PI/2 radians. *
 * Special cases:
 * - `asin(x)` is `NaN`, when `abs(x) > 1` or `x` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun asin(x: Double): Double = nativeMath.asin(x)

/** Computes the arc cosine of the value [x]; the returned value is an angle in the range from 0.0 to PI radians. *
 * Special cases:
 * - `acos(x)` is `NaN`, when `abs(x) > 1` or `x` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun acos(x: Double): Double = nativeMath.acos(x)

/** Computes the arc tangent of the value [x]; the returned value is an angle in the range from -PI/2 to PI/2 radians. *
 * Special cases:
 * - `atan(x)` is `NaN`, when `x` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun atan(x: Double): Double = nativeMath.atan(x)

/** Computes the arc tangent of the value [x] and [y] by computing the arc tangent of the value [y] / [x]; the returned value is an angle in the range from -PI to PI radians. *
 * Special cases:
 * - `atan2(0.0, 0.0)` is `0.0`
 */
@SinceKotlin("1.2")
public actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x)

/** Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x]; the returned value is an angle in the range from -PI to PI radians. *
 * Special cases:
 * - `atan2(0.0, 0.0)` is `0.0`
 */
@SinceKotlin("1.2")
public actual inline fun atan2(y: Double, x: Double): Double = nativeMath.atan2(y, x)

/** Computes the hyperbolic sine of the value [x]. *
 * Special cases:
 * - `sinh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun sinh(x: Double): Double = nativeMath.sinh(x)

/** Computes the hyperbolic cosine of the value [x]. *
 * Special cases:
 * - `cosh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun cosh(x: Double): Double = nativeMath.cosh(x)

/** Computes the hyperbolic tangent of the value [x]. *
 * Special cases:
 * - `tanh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)

/** Computes the inverse hyperbolic sine of the value [x]. *
 * Special cases:
 * - `asinh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun asinh(x: Double): Double = nativeMath.asinh(x)

/** Computes the inverse hyperbolic cosine of the value [x]. *
 * Special cases:
 * - `acosh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun acosh(x: Double): Double = nativeMath.acosh(x)

/** Computes the inverse hyperbolic tangent of the value [x]. *
 * Special cases:
 * - `tanh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun tanh(x: Double): Double = nativeMath.tanh(x)

/** Computes the inverse hyperbolic tangent of the value [x]. *
 * Special cases:
 * - `tanh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun asinh(x: Double): Double = nativeMath.asinh(x)

/** Computes the inverse hyperbolic cosine of the value [x]. *
 * Special cases:
 * - `acosh(NaN)` is `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun acosh(x: Double): Double = nativeMath.acosh(x)
Double): Double = nativeMath.hypot(x, y)\n\n* Computes the positive square root of the value [x].\n* Special cases:\n  * - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN``\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* sqrt(x: Double): Double = nativeMath.sqrt(x)\n
/**
 * Computes Euler's number `e` raised to the power of the value [x].\n* Special cases:\n  * - `exp(NaN)` is `NaN``\n  * - `exp(±Inf)` is `±Inf``\n  * - `exp(-Inf)` is `-0.0``\n*/
* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* exp(x: Double): Double = nativeMath.exp(x)\n
/**
 * Computes `exp(x) - 1`.\n* This function can be implemented to produce more\n* precise result for [x] near zero.\n* Special cases:\n  * - `expm1(NaN)` is `NaN``\n  * - `expm1(±Inf)` is `±Inf``\n  * - `expm1(-Inf)` is `-1.0``\n* @see [exp] function.\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* expm1(x: Double): Double = nativeMath.expm1(x)\n
/**
 * Rounds the given value [x] to an integer towards positive infinity.\n* @return the\n* smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* ceil(x: Double): Double = nativeMath.ceil(x).unsafeCast<Double>()\n
/**
 * Rounds the given value [x] to an integer towards negative infinity.\n* @return the\n* largest double value that is smaller than or equal to the given value [x] and is a mathematical integer.\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* floor(x: Double): Double = nativeMath.floor(x).unsafeCast<Double>()\n
/**
 * Rounds the given value [x] to an integer towards zero.\n* @return the\n* value [x] having its fractional part truncated.\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* trunc(x: Double): Double = nativeMath.trunc(x)\n
/**
 * Returns the absolute value of the\n* given value [x].\n* @return the\n* absolute value of the given value [x].\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* abs(x: Double): Double = nativeMath.abs(x)\n
/**
 * Returns the sign of the given value [x]:\n*   - `-1.0` if the value is negative,\n* @SinceKotlin("1.2")\n* @InlineOnly\n* @public\n* actual\n* inline\n* fun\n* sign(x: Double): Double =
Float Math

---

*Computes the sine of the angle [x] given in radians.*

```
*atSin(NaN) = NaN
```

*Computes the cosine of the angle [x] given in radians.*

```
*atCos(NaN) = NaN
```

*Computes the tangent of the angle [x] given in radians.*

```
*atTan(NaN) = NaN
```

---

Float Math

---

*Computes the arc sine of the value [x] in radians.*

```
*asSin(NaN) = NaN
```

*Computes the arc cosine of the value [x] in radians.*

```
*asCos(NaN) = NaN
```

*Computes the arc tangent of the value [x] in radians.*

```
*atTan(NaN) = NaN
```

---

Float Math

---

*Computes the square root of the value [x].*  

```
*sqrt(NaN) = NaN
```

---

Float Math

---

*Computes the natural logarithm of the value [x].*  

```
*log(NaN) = NaN
```

---

Float Math

---

*Computes the exponential of the value [x].*  

```
*exp(NaN) = NaN
```

---

Float Math

---

*Computes the logarithm base 2 of the value [x].*  

```
*log2(NaN) = NaN
```

---

Float Math

---

*Computes the logarithm base e of the value [x].*  

```
*log10(NaN) = NaN
```

---

Float Math

---

*Computes the inverse hyperbolic sine of the value [x].*  

```
*asinh(NaN) = NaN
```

---

Float Math

---

*Computes the inverse hyperbolic cosine of the value [x].*  

```
*acosh(NaN) = NaN
```

---

Float Math

---

*Computes the inverse hyperbolic tangent of the value [x].*  

```
*tanh(NaN) = NaN
```

---

Float Math

---

*Computes the square root of the value [x].*  

```
*sqrt(x) = x
```

---

Float Math

---

*Computes the natural logarithm of the value [x].*  

```
*log(x) = x
```

---

Float Math

---

*Computes the exponential of the value [x].*  

```
*exp(x) = x
```

---

Float Math

---

*Computes the logarithm base 2 of the value [x].*  

```
*log2(x) = x
```

---

Float Math

---

*Computes the logarithm base e of the value [x].*  

```
*log10(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic sine of the value [x].*  

```
*asinh(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic cosine of the value [x].*  

```
*acosh(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic tangent of the value [x].*  

```
*tanh(x) = x
```

---

Float Math

---

*Computes the square root of the value [x].*  

```
*sqrt(x) = x
```

---

Float Math

---

*Computes the natural logarithm of the value [x].*  

```
*log(x) = x
```

---

Float Math

---

*Computes the exponential of the value [x].*  

```
*exp(x) = x
```

---

Float Math

---

*Computes the logarithm base 2 of the value [x].*  

```
*log2(x) = x
```

---

Float Math

---

*Computes the logarithm base e of the value [x].*  

```
*log10(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic sine of the value [x].*  

```
*asinh(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic cosine of the value [x].*  

```
*acosh(x) = x
```

---

Float Math

---

*Computes the inverse hyperbolic tangent of the value [x].*  

```
*tanh(x) = x
```

---

Float Math

---

*Computes the square root of the value [x].*  

```
*sqrt(x) = x
```

---

Float Math

---

*Computes the natural logarithm of the value [x].*  

```
*log(x) = x
```

---

Float Math

---

*Computes the exponential of the value [x].*  

```
*exp(x) = x
```

---

Float Math

---

*Computes the logarithm base 2 of the value [x].*  

```
*log2(x) = x
```

---

Float Math

---

*Computes the logarithm base e of the value [x].*  

```
*log10(x) = x
```
hypot(x: Float, y: Float): Float = nativeMath.hypot(x.toDouble(), y.toDouble()).toFloat()

/**
 * Computes the positive square root of the value [x] if either `x` or `y` are `NaN`. `hypot(1.0, 1.0)` is `1.4142135623730951`.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun hypot(x: Float, y: Float): Float = nativeMath.hypot(x.toDouble(), y.toDouble()).toFloat()

/**
 * Computes Euler's number `e` raised to the power of the value `x`.
 *
 * Special cases:
 *   - `exp(x)` is `NaN` when `x` is `NaN`.
 *   - `exp(-Inf)` is `0.0`.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun exp(x: Float): Float = exp(x.toDouble()).toFloat()

/**
 * Computes the logarithm of the value `x` to the given base.
 *
 * Special cases:
 *   - `log(x)` is `NaN` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.
 *   - `log(-0)` is `-Inf`.
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log(x: Float, base: Float): Float = log(x.toDouble(), base.toDouble()).toFloat()
nativeMath.sign(x.toDouble()).toFloat()\n
/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public actual inline fun min(a: Float, b: Float): Float = nativeMath.min(a, b)\n
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public actual inline fun max(a: Float, b: Float): Float = nativeMath.max(a, b)\n
/**
 * Returns the absolute value of this value.
 */
@SinceKotlin("1.2")
public actual inline val Float.absoluteValue: Float get() = nativeMath.abs(this.toDouble()).toFloat()\n
/**
 * Returns the sign of this value:
 *   - `-1.0` if the value is negative,
 *   - zero if the value is zero,
 *   - `1.0` if the value is positive
 */
@SinceKotlin("1.2")
public actual inline val Float.sign: Float get() = nativeMath.sign(this.toDouble()).toFloat()\n
/**
 * Returns this value with the sign bit same as of the [sign] value.
 */
@SinceKotlin("1.2")
public actual inline fun Float.withSign(sign: Float): Float = this.toDouble().withSign(sign.toDouble()).toFloat()\n
/**
 * Returns this value with the sign bit same as of the [sign] value.
 */
@SinceKotlin("1.2")
public actual inline fun Float.withSign(sign: Int): Float = this.toDouble().withSign(sign.toDouble()).toFloat()\n
/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 */
@SinceKotlin("1.2")
public actual inline fun Float.roundToInt(): Int = toDouble().roundToInt()\n
/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Long].
 * Ties are rounded towards positive infinity.
 */
@SinceKotlin("1.2")
public actual inline fun Float.roundToLong(): Long = toDouble().roundToLong()\

/**
 * Returns the absolute value of the given value.
 */
@SinceKotlin("1.2")
public actual fun abs(n: Int): Int = if (n < 0) (-n or 0) else n\n
/**
 * Returns the absolute value of the given value.
 */
@SinceKotlin("1.2")
public actual fun abs(n: Long): Long = if (n < 0) -n else n\n
====================================================================
public actual inline fun min(a: Long, b: Long): Long = if (a <= b) a else b

/**
 * Returns the greater of two values.
 */
public actual inline fun max(a: Long, b: Long): Long = if (a >= b) a else b

/**
 * Returns the absolute value of this value.
 *
 * Special cases:
 * - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE` due to an overflow
 * - `abs(0)` is `0`
 *
 * @see abs function
 */
public actual inline fun Long.absoluteValue: Long get() = abs(this)

/**
 * Returns `true` if the specified number is a Not-a-Number (NaN) value, `false` otherwise.
 */
public actual fun Double.isNaN(): Boolean = this != this

/**
 * Returns `true` if the specified number is a Not-a-Number (NaN) value, `false` otherwise.
 */
public actual fun Float.isNaN(): Boolean = this != this

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Double.isFinite(): Boolean = !isInfinite() && !isNaN()

/**
 * Returns `true` if the argument is a finite floating-point value; returns `false` otherwise (for `NaN` and infinity arguments).
 */
public actual fun Float.isFinite(): Boolean = !isInfinite() && !isNaN()

/**
 * Counts the number of set bits in the binary representation of this [Int] number.
 */
public actual fun Int.countOneBits(): Int {
    var v = this
    v = (v and 0x55555555) + (v.ushr(1) and 0x55555555)
    v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)
    v = (v and 0x0F0F0F0F) + (v.ushr(4) and 0x0F0F0F0F)
    v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)
    return v
}

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */
public actual fun Int.countLeadingZeroBits(): Int = kotlin.js.Math.clz32(this)

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.
 */
public actual fun Int.countTrailingZeroBits(): Int = Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number.
 */
public actual fun Int.takeHighestOneBit(): Int = 

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number.
 */
public actual fun Int.takeLowestOneBit(): Int =

/**
 * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
 */
public actual fun Int.rotateLeft(bitCount: Int): Int =

/**
 * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.
 */
public actual fun Int.rotateRight(bitCount: Int): Int =

/**
 * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
 */
public actual fun Int.rotateLeft(bitCount: Int): Int =

/**
 * Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.
 */
public actual fun Int.rotateRight(bitCount: Int): Int =

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */
the number as the most significant bits on the left side.

*\n* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.

\`
number.rotateRight(-n) == number.rotateLeft(n)
\`

*\n* Rotating by a multiple of `Int.SIZE_BITS` (32) returns the same number, or more generally:

\`
\text{number.rotateRight(n)} == number.rotateRight(n \mod 32)
\`

*/

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.rotateRight(bitCount: Int): Int =
    shl(Int.SIZE_BITS - bitCount) or ushr(bitCount)

/**
 * Counts the number of set bits in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.countOneBits(): Int =
    high.countOneBits() + low.countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.countLeadingZeroBits(): Int =
    when (val high = this.high) {
        0 -> Int.SIZE_BITS + low.countLeadingZeroBits()
        else -> high.countLeadingZeroBits()
    }

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.countTrailingZeroBits(): Int =
    when (val low = this.low) {
        0 -> Int.SIZE_BITS + high.countTrailingZeroBits()
        else -> low.countTrailingZeroBits()
    }

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Int] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.takeHighestOneBit(): Int =
    when (val high = this.high) {
        0 -> Int(low.takeHighestOneBit(), 0)
        else -> Long(high, low.takeHighestOneBit())
    }

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Int] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.takeLowestOneBit(): Int =
    when (val low = this.low) {
        0 -> Long(0, high.takeLowestOneBit())
        else -> Long(high, low.takeLowestOneBit())
    }

/**
 * Rotates the binary representation of this [Long] number right by a negative bit count is the same as rotating it left by the negated bit count:

\`
number.rotateRight(-n) == number.rotateLeft(n)
\`

Rotating by a multiple of `Long.SIZE_BITS` (64) returns the same number, or more generally:

\`
\text{number.rotateRight(n)} == number.rotateRight(n \mod 64)
\`

*/

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateRight(bitCount: Int): Long =
    if ((bitCount and 31) != 0) {
        val low = this.low
        val high = this.high
        val newLow =
            low.shl(bitCount) or high.ushr(-bitCount)
        val newHigh = high.shl(bitCount) or low.ushr(-bitCount)
        return if ((bitCount and 32) == 0) Long(newLow, newHigh) else Long(high, low)
    } else {
        return if ((bitCount and 32) == 0) this else Long(high, low)
    }

/**
 * Rotates the binary representation of this [Long] number right by a negative bit count is the same as rotating it left by the negated bit count:

\`
number.rotateRight(-n) == number.rotateLeft(n)
\`

Rotating by a multiple of `Long.SIZE_BITS` (64) returns the same number, or more generally:

\`
\text{number.rotateRight(n)} == number.rotateRight(n \mod 64)
\`

*/

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateRight(bitCount: Int): Long =
    rotateLeft(-bitCount)

/**
 * Exposes the JavaScript [Promise object](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise) to Kotlin.
 */
@Suppress("NOT_DOCUMENTED")
public final class Promise<out T>(executor: (resolve: (T) -> Unit, reject: (Throwable) -> Unit) -> Unit) {
    public open fun <S>
        then(onFulfilled: ((T) -> S)?): Promise<S>

    public open fun <S>
        then(onFulfilled: ((T) -> S)?, onRejected: ((Throwable) -> S)?): Promise<S>

    public open fun <S>
catch(onRejected: (Throwable) -> S): Promise<S>?

    companion object {
        public fun <S> all(promise: Array<out Promise<S>>): Promise<Array<out S>>{
            return Promise.all(promise.map { it }).then { result: S[] -> result }
        }

        public fun <S> race(promise: Array<out Promise<S>>): Promise<S>{
            return Promise.race(promise.map { it }).then { result: S -> result }
        }

        public fun reject(e: Throwable): Promise<Nothing>{
            return Promise.reject(e)
        }

        public fun <S> resolve(e: S): Promise<S>{
            return Promise.resolve(e)
        }
    }
}

// It's workaround
inline fun <T, S> Promise<Promise<T>>.then(
    noinline onFulfilled: ((T) -> S)?,
    noinline onRejected: ((Throwable) -> S)?
): Promise<S> {
    return this.unsafeCast<Promise<T>>().then(onFulfilled, onRejected)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.
 */

package kotlin.random

import kotlin.math.pow

internal actual fun defaultPlatformRandom(): Random =
    Random(js("(Math.random() * Math.pow(2, 32)) | 0")).unsafeCast<Int>()

private val INV_2_26: Double = 2.0.pow(-26)
private val INV_2_53: Double = 2.0.pow(-53)
internal actual fun doubleFromParts(hi26: Int, low27: Int): Double =
    hi26 * INV_2_26 + low27 * INV_2_53

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.js

import kotlin.reflect.KClass
import kotlin.reflect.js.internal.KClassImpl

/**
 * Represents the constructor of a class. Instances of `JsClass` can be
 * passed to JavaScript APIs that expect a constructor reference.
 */
external interface JsClass<T : Any> {
    /**
     * Returns the unqualified name of the class represented by this instance.
     */
    val name: String
}

@Deprecated("Use class literal and extension property `js` instead.", replaceWith =
    ReplaceWith("T::class.js"), level = DeprecationLevel.ERROR)
external fun <T : Any> jsClass(): JsClass<T>

@Deprecated("Use class literal and extension property `js` instead.", replaceWith =
    ReplaceWith("this::class.js"), level = DeprecationLevel.ERROR)
val <T : Any> T.jsClass: JsClass<T>
    get() = when (jsTypeOf(this)) {
        "string" -> js("String")
        "number" -> js("Number")
        "boolean" -> js("Boolean")
        else -> js("Object").getPrototypeOf(this).constructor
    }

/**
 * Obtains a `KClass` instance for the given constructor reference.
 */
val <T : Any> KClass<T>.js: JsClass<T>
    get() = (this as KClassImpl<T>).jClass

/**
 * Obtains a constructor reference for the given `KClass`.
 */
val <T : Any> KClass<T>.js: JsClass<T>
    get() = getKClass(this)

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal abstract class KClassImpl<T : Any>(
    internal open val jClass: JsClass<T>
) : KClass<T> {
    override val annotations: List<Annotation> = TODO()
    override val constructors: Collection<KFunction<T>> = TODO()
    override val isAbstract: Boolean = TODO()
    override val isCompanion: Boolean = TODO()
    override val isData: Boolean = TODO()
    override val isFinal: Boolean = TODO()
    override val isOpen: Boolean = TODO()
    override val isSealed: Boolean = TODO()
    override val members: Collection<KCallable<*>> = TODO()
    override val nestedClasses: Collection<KClass<*>> = TODO()
    override val objectInstance: T? = TODO()
    override val qualifiedName: String? = TODO()
    override val supertypes: List<KType> = TODO()
    override val typeParameters: List<KTypeParameter> = TODO()
    override val sealedSubclasses: List<KClass<out T>> = TODO()
    override val visibility: KVisibility? = TODO()

    override fun equals(other: Any?): Boolean {
        return other is KClassImpl<*> && jClass == other.jClass
    }

    override fun hashCode(): Int = jClass.hashCode()

    override fun toString(): String = jClass
}

internal class SimpleKClassImpl<T : Any>(jClass: JsClass<T>) : KClassImpl<T>(jClass) {
    override val simpleName: String? = jClass.asDynamic().`$metadata$`?.simpleName

    override fun isInstance(value: Any?): Boolean {
        return jsIsType(value, jClass)
    }
}
PrimitiveKClassImpl<T : Any>(
    jClass: JsClass<T>,
    private val givenSimpleName: String,
    private val isInstanceFunction: (Any?) -> Boolean
) : KClassImpl<T>(jClass) {
    override fun equals(other: Any?): Boolean {
        if (other !is PrimitiveKClassImpl<*>) return false
        return super.equals(other) && givenSimpleName == other.givenSimpleName
    }

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override fun simpleName: String? get() = givenSimpleName
}

internal object NothingKClassImpl : KClassImpl<Nothing>(js("Object")) {
    override val simpleName: String = "Nothing"

    override fun isInstance(value: Any?): Boolean = false

    override fun isInstance(value: Any?): Boolean = false

    override fun jClass: JsClass<Nothing> = throw UnsupportedOperationException("There's no native JS class for Nothing type")

    override fun isInstance(value: Any?): Boolean = false

    override fun isInstance(value: Any?): Boolean = false

    override fun equals(other: Any?): Boolean = other.isInstanceFunction(value)

    override fun hashCode(): Int = 0
}

import kotlin.reflect.*

internal class KTypeImpl(
    override val classifier: KClassifier,
    override val arguments: List<KTypeProjection>,
    override val isMarkedNullable: Boolean
) : KType {
    override val annotations: List<Annotation> = emptyList()

    override fun equals(other: Any?): Boolean {
        return other is KTypeImpl &&
                classifier == other.classifier && arguments == other.arguments && isMarkedNullable == other.isMarkedNullable
    }

    override fun hashCode(): Int =
        (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode()

    override fun toString(): String {
        val kClass = (classifier as? KClass<*>)

        when {
            kClass == null -> classifier.toString()
            kClass.simpleName != null
            -> kClass.simpleName
            else -> "(non-denotable type)"
        }

        val args =
            if (arguments.isEmpty()) ""
            else arguments.joinToString(", ", "<", ">") { it.asString() }
        val nullable = if (isMarkedNullable) "?" else ""

        return classifierName + args + nullable
    }
}

internal object DynamicKType : KType {
    override val classifier: KClassifier? = null
    override val arguments: List<KTypeProjection> = emptyList()
    override val isMarkedNullable: Boolean = false
    override val annotations: List<Annotation> = emptyList()
    override fun toString(): String = "dynamic"
}

internal fun KVariance.prefixString() =
    when (this) {
        KVariance.INVARIANT -> "V"
        KVariance.IN -> "V" + (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode
        KVariance.OUT -> "V" + (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode
        KVariance.OUTVARIANT -> "V" + (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode
        KVariance.INVARIANT -> "V" + (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode
    }

// TODO: Remove once JsReflectionAPICallChecker supports more reflection types

package kotlin.reflect.js.internal

import kotlin.reflect.*

internal class KTypeImpl
    classifier: KClassifier,
    arguments: List<KTypeProjection>,
    isMarkedNullable: Boolean
) : KType {
    override val annotations: List<Annotation> = emptyList()

    override fun equals(other: Any?): Boolean {
        return other is KTypeImpl &&
                classifier == other.classifier && arguments == other.arguments && isMarkedNullable == other.isMarkedNullable
    }

    override fun hashCode(): Int =
        (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
        isMarkedNullable.hashCode

    override fun toString(): String {
        val kClass = (classifier as? KClass<*>)

        when {
            kClass == null -> classifier.toString()
            kClass.simpleName != null
            -> kClass.simpleName
            else -> "(non-denotable type)"
        }

        val args =
            if (arguments.isEmpty()) ""
            else arguments.joinToString(", ", "<", ">") { it.asString() }
        val nullable = if (isMarkedNullable) "?" else ""

        return classifierName + args + nullable
    }
}

internal object DynamicKType : KType {
    override val classifier: KClassifier? = null
    override val arguments: List<KTypeProjection> = emptyList()
    override val isMarkedNullable: Boolean = false
    override val annotations: List<Annotation> = emptyList()
    override fun toString(): String = "dynamic"

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override fun equals(other: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }

    override fun jClass: JsClass<Nothing> = throw UnsupportedOperationException("There's no native JS class for Nothing type")

    override fun isInstance(value: Any?): Boolean = false

    override fun isInstance(value: Any?): Boolean = false

    override fun equals(other: Any?): Boolean = other.isInstanceFunction(value)

    override fun hashCode(): Int = 0
}
arrayOfNulls<KClassImpl<Any>>(0), /*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
 * Language contributors. */ Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

// a package is omitted to get declarations directly under the module

import kotlin.reflect.*
import kotlin.reflect.js.internal.*

@JsName("getKClass")
internal fun <T : Any> getKClass(jClass: JsClass<T>): KClass<T> =
    getOrCreateKClass(jClass)

@JsName("getKClassFromExpression")
internal fun <T : Any> getKClassFromExpression(e: T): KClass<T> =
    when (jsTypeOf(e)) {
        "string" ->
            PrimitiveClasses.stringClass
        "number" -> if (jsBitwiseOr(e, 0).asDynamic() === e)
            PrimitiveClasses.doubleClass
        "boolean" -> PrimitiveClasses.booleanClass
    }

private fun <T : Any> getOrCreateKClass(jClass: JsClass<T>): KClass<T> {
    if (jClass === js("String")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()

    val metadata = jClass.asDynamic().`$metadata$`
    return if (metadata != null) {
        if (metadata.`$kClass$` == null) {
            val kClass = SimpleKClassImpl(jClass)
            metadata.`$kClass$` = kClass
            kClass
        } else {
            metadata.`$kClass$`
        }
    } else {
        SimpleKClassImpl(jClass)
    }
}

package kotlin.js

/**
 */
@Suppress("NOT_DOCUMENTED")
public external class RegExp(pattern: String, flags: String? = definedExternally) {

    public fun test(str: String): Boolean

    public fun exec(str: String): RegExpMatch?

    public override fun toString(): String

    /**
     * The lastIndex is a read/write integer property of regular
     * expressions that specifies the index at which to start the next match.
     */
    public var lastIndex: Int

    public var global: Boolean
    public var ignoreCase: Boolean
    public var multiline: Boolean
}

/**
 * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning
 * of the input string.
 */
public fun RegExp.reset() {
    lastIndex = 0
}

// TODO: Inherit from array or
 /**
 * Represents the return value of [RegExp.exec].
 */
@Suppress("NOT_DOCUMENTED")
public external interface RegExpMatch {
    public val index: Int

    public val input: String

    public val length: Int

    /**
     * Returns the entire text matched by [RegExp.exec] if
     * the [index] parameter is 0, or the text matched by each capturing parenthesis
     * at the given index.
     */

    /**
     * Converts the result of [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent
     * element is the text matched by each capturing parenthesis.
     */
    public inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()
}

package kotlin.sequences

internal actual class ConstrainedOnceSequence<T> actual constructor(sequence: Sequence<T>) : Sequence<T> {
    private var sequenceRef: Sequence<T>? = sequence

    actual override fun iterator(): Iterator<T> {
        val sequence =
@param startIndex the beginning (inclusive) of the subsequence to append.
 * @param endIndex the end (exclusive) of the subsequence to append.
 * @throws IndexOutOfBoundsException or [IllegalStateException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @param SinceKotlin("1.3")
 * @ExperimentalStdlibApi
 * @SinceKotlin("1.3")
 * @PublicFun appendRange(value: CharSequence?, startindex: Int, endindex: Int): StringBuilder {
 * @param value the array from which characters are inserted.
 * @param startindex the beginning (inclusive) of the subarray to insert.
 * @param endindex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * @Throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException if [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @Throws IndexOutOfBoundsException if [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @Throws IndexOutOfBoundsException if [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @param startindex the beginning (inclusive) of the subsequence to append.
 * @param endindex the end (exclusive) of the subsequence to append.
 * @Throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @param value the character sequence from which a subsequence is inserted. If [value] is `null`, then characters will be inserted as if [value] contained the four characters `\"null\"`.
 * @param startindex the beginning (inclusive) of the subsequence to insert.
 * @param endindex the end (exclusive) of the subsequence to insert.
 * @Throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subsequence to append.
 * @param endIndex the end (exclusive) of the subsequence to append.
 * @param value the character sequence from which a subsequence is inserted. If [value] is `null`, then characters will be inserted as if [value] contained the four characters `\"null\"`.
part of a supplementary code point, this method does not remove the entire supplementary character.\n
@throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.\n
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.deleteAt(index: Int): StringBuilder =
    this.deleteAt(index)\n
/**
 * Removes characters in the specified range from this string builder and returns this instance.
 * @param startIndex the beginning (inclusive) of the range to remove.
 * @param endIndex the end (exclusive) of the range to remove.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE", "ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")\npublic actual inline fun StringBuilder.deleteRange(startIndex: Int, endIndex: Int): StringBuilder =
    this.deleteRange(startIndex, endIndex)\n
/**
 * Copies characters from this string builder into the [destination] character array.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
 * @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the string builder indices or when `startIndex > endIndex`.
 * @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) =
    this.toCharArray(destination, destinationOffset, startIndex, endIndex)\n
/**
 * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.
 * Characters are appended in order, starting at specified [startIndex].
 * @param value the array from which characters are appended.
 * @param startIndex the beginning (inclusive) of the subarray to append.
 * @param endIndex the end (exclusive) of the subarray to append.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.appendRange(value: CharArray?, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)\n
/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.appendRange(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)\n
/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * @param index the position in this string builder to insert at.
 * @param value the array from which characters are inserted.
 * @param startIndex the beginning (inclusive) of the subarray to insert.
 * @param endIndex the end (exclusive) of the subarray to insert.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@Suppress("EXTENSION_SHADOWED_BY_MEMBER", "NOTHING_TO_INLINE")\npublic actual inline fun StringBuilder.appendRange(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)

Open Source Used In DNAC Platform Arica  13345
length of this string builder."

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

// actually \s is enough to match all whitespace, but \xA0 added because of different regexp behavior of Rhino used in Selenium tests
public actual fun Char.isWhitespace(): Boolean = toString().matches("[\s\xA0]")

@kotlin.internal.InlineOnly
public actual inline fun Char.toLowerCase(): Char = js("String.fromCharCode")((toInt()).toLowerCase().charCodeAt(0).unsafeCast<Int>().toChar())

@kotlin.internal.InlineOnly
public actual inline fun Char.toUpperCase(): Char = js("String.fromCharCode")((toInt()).toUpperCase().charCodeAt(0).unsafeCast<Int>().toChar())

/**
 * Returns `true` if this character is a Unicode high-surrogate code unit (also known as leading-surrogate code unit).
 */
public actual fun Char.isHighSurrogate(): Boolean = this in Char.MIN_HIGH_SURROGATE..Char.MAX_HIGH_SURROGATE

/**
 * Returns `true` if this character is a Unicode low-surrogate code unit (also known as trailing-surrogate code unit).
 */
public actual fun Char.isLowSurrogate(): Boolean = this in Char.MIN_LOW_SURROGATE..Char.MAX_LOW_SURROGATE

/**
 * Returns `true` if the contents of this string is equal to the word "true", ignoring case, and `false` otherwise.
 */
public actual fun String.toBoolean(): Boolean = toLowerCase() == "true"

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toByte(): Byte = toByteOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a signed [Byte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
public actual fun String.toByte(radix: Int): Byte = toByteOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an [Int] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toInt(): Int = toIntOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [Int] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toInt(radix: Int): Int = toIntOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as a [Short] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
public actual fun String.toShort(radix: Int): Short = toShortOrNull(radix) ?: numberFormatError(this)
numberFormatError(this)\n\n/**
 * Parses the string as an [Int] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */

public actual fun Int.toIntOrNull(radix: Int): Int = toInt(radix) ?: numberFormatError(this)\n\n/**
 * Parses the string as a [Double] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */

public actual fun String.toDouble(): Double = (+(this.asDynamic())).unsafeCast<Double>().also {\n    if (it.isNaN() && it == 0.0 \n        || this.isBlank())\n          numberFormatError(this)\n}\n\n/**
 * Parses the string as a [Double] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */

public actual fun String.toDoubleOrNull(): Double? = (+(this.asDynamic())).unsafeCast<Double>().takeIf {\n    !(it.isNaN() && this.isBlank())\n}\n\n/**
 * Parses the string as a [Byte] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */

public actual inline fun Byte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**
 * Provides enumeration values to use to set regular expression options.
 */

public actual enum class RegexOption(val value: String) {
    /** Enables case-insensitive matching. */
    IGNORE_CASE("i"),
    /** Enables multiline mode. */
    MULTILINE("m")
}

/**
 * Represents a compiled regular expression.
 * Provides functions to match strings in text with a pattern, replace the found occurrences and split text around matches.
 */

public actual class Regex(internal fun toString(radix: Int): String = this.toInt().toString(radix))\n\n/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 */

public actual fun Byte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**
 * Represents a compiled regular expression.
 * Provides functions to match strings in text with a pattern, replace the found occurrences and split text around matches.
 */

public actual class Regex(internal fun toString(radix: Int): String = this.toInt().toString(radix))\n\n/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 */

public actual fun Byte.toString(radix: Int): String = this.toInt().toString(radix)\n\n/**
 * Provides enumeration values to use to set regular expression options.
 */

public actual enum class RegexOption(val value: String) {
    /** Enables case-insensitive matching. */
    IGNORE_CASE("i"),
    /** Enables multiline mode. */
    MULTILINE("m")
}
public actual class Regex constructor(pattern: String, options: Set<RegexOption>) {

    public actual constructor(pattern: String, option: RegexOption) : this(pattern, setOf(option))

    public actual constructor(pattern: String) : this(pattern, emptySet())

    val pattern: String = pattern
    val options: Set<RegexOption> = options.toSet()
    private val nativePattern: RegExp = RegExp(pattern, options.map {
        it.value
    }.joinToString(separator = "") + "g")

    infix fun matches(input: CharSequence): Boolean {
        nativePattern.reset()
        val match = nativePattern.exec(input.toString())
        return match != null && match.index == 0 &&
        nativePattern.lastIndex == input.length
    }

    fun containsMatchIn(input: CharSequence): Boolean {
        nativePattern.reset()
        return nativePattern.test(input.toString())
    }

    fun find(input: CharSequence, startIndex: Int = 0): MatchResult? = nativePattern.findNext(input.toString(), startIndex)

    fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> =
        generateSequence({ find(input, startIndex) }, { match -> match.next() })

    fun matchEntire(input: CharSequence): MatchResult? {
        if (pattern.startsWith('^') &&
            pattern.endsWith('$'))
            return find(input)
        else
            return

    }

    fun replace(input: CharSequence, replacement: String): String = input.toString().nativeReplace(nativePattern, replacement)

    fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String {
        val match = find(input)
        if (match == null) return

        val lastStart = 0
        val length = input.length
        val sb = StringBuilder(length)
        do {
            val foundMatch = match!!
            sb.append(input, lastStart, foundMatch.range.start)
            sb.append(transform(foundMatch))
            lastStart = foundMatch.range.endInclusive + 1
            match = foundMatch.next()
        } while (lastStart < length && match != null)

        return sb.toString()
    }

    fun replaceFirst(input: CharSequence, replacement: String): String {
        val nonGlobalOptions =
options.map { it.value } joinToString(separator = "\")
        return input.toString().nativeReplace(RegExp(pattern,
            nonGlobalOptions), replacement) /* Splits the [input] CharSequence around matches of this
regular expression. */ @param limit Non-negative value specifying the maximum number of substrings
the string can be split to. */ * Zero by default means no limit is set. */ @param pattern Regular expression to search for in the [input] string. */ @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS") public actual fun split(input:
        CharSequence, limit: Int = 0): List<String> { var lastStart = 0
        for (match in matches) { val result = mutableListOf<String>()
            result.add(input.subSequence(lastStart, match.range.start).toString())
            lastStart = match.range.endInclusive + 1
        }
        result.add(input.subSequence(lastStart, input.length).toString())
        return result } /* Returns the string representation of this regular expression, namely the [pattern] of this regular
expression. */ Note that another regular expression constructed from the same pattern string may have
different [options] */ * and may match strings differently. */ public override fun toString(): String =
        nativePattern.toString() /* Returns a regular expression pattern string that matches the specified [literal] string literally. */ * No characters of that string will have special meaning when searching for an occurrence of the regular
expression. */ * public actual fun fromLiteral(literal: String): Regex = Regex(escape(literal)) /* Returns a literal replacement expression for the
specified [literal] string. */ * No characters of that string will have special meaning when it is used as a
replacement string in [Regex.replace] function. */ * public actual fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, "$$") Creates a regular expression from the specified [pattern] string and the specified single [option]. */ *@Deprecated("Provided for binary compatibility ", level = DeprecationLevel.HIDDEN) */ *@JsName("Regex_sb3q2\$") */ public fun Regex_0(pattern: String, option:
        RegexOption): Regex = Regex(pattern, setOf(option)) /* Creates a regular expression from the specified [pattern] string and the default options. */ @Deprecated("Provided for binary compatibility ", level = DeprecationLevel.HIDDEN) */ @JsName("Regex_61zpoe\$") */ public fun Regex_1(pattern: String): Regex =
        Regex(pattern, emptySet()) /* Returns the string representation of this regular expression, namely the [pattern] of this regular
expression. */ override fun get(index: Int): MatchResult? = match[0]!!
        override val value: String = match[0] ?: ""
        override fun next(): MatchResult? = this@findNext.findNext(input, fromIndex)
*/ return object : MatchResult {
            override val range: IntRange = range
            override val value: String
                get() = match[0]!!
            override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it] }.iterator()
                override fun get(index: Int): MatchGroup? = match[index]?
                override val size: Int get() = match.length
                override fun get(index: Int): String? = match[index]
                override fun get(index: Int): MatchGroup? = this@findNext.findNext(input, range)
                override fun replace(input: String, replacement: String): String =
                override fun replaceAll(pattern: String, replacement: String): String =
                override val groups: MatchGroupCollection = object : MatchGroupCollection, AbstractCollection<MatchGroup?> {
                    override fun size(): Int = match.length
                    override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it] }.iterator()
                    override fun get(index: Int): MatchGroup? = match[index]?
                    override val size: Int get() = match.length
                    override fun get(index: Int): String = match[index]
                    override fun get(index: Int): MatchGroup? = this@findNext.findNext(input, range)
                    override fun replaceAll(pattern: String, replacement: String): String =
                    override fun replaceAll(pattern: String, replacement: String): String =
                    override val groups: MatchGroupCollection = object : MatchGroupCollection, AbstractCollection<MatchGroup?> {
                        override fun size(): Int = match.length
                        override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it] }.iterator()
                        override fun get(index: Int): MatchGroup? = match[index]?
                        override val size: Int get() = match.length
                        override fun get(index: Int): String = match[index]
                        override fun get(index: Int): MatchGroup? = this@findNext.findNext(input, range)
                        override fun replaceAll(pattern: String, replacement: String): String =
                        override fun replaceAll(pattern: String, replacement: String): String =
                        override val groups: MatchGroupCollection = object : MatchGroupCollection, AbstractCollection<MatchGroup?> {
                            override fun size(): Int = match.length
                            override fun iterator(): Iterator<MatchGroup?> = indices.asSequence().map { this[it] }.iterator()
license/LICENSE.txt file:

```kotlin
package kotlin.text

/** Returns the negative [size] if [throwOnMalformed] is false, throws [CharacterCodingException] otherwise.
* @UseExperimental(ExperimentalStdlibApi::class)
*/
private fun malformed(size: Int, index: Int, throwOnMalformed: Boolean): Int {
    if (throwOnMalformed) throw CharacterCodingException("Malformed sequence starting at ${index - 1}")
    return -size
}

/**
 * Returns code point corresponding to UTF-16 surrogate pair,[n] * where the first of the pair is the [high] and the second is in the [string] at the [index].
* Returns zero if the pair is malformed and [throwOnMalformed] is false.
* @throws CharacterCodingException if the pair is malformed and [throwOnMalformed] is true.
* @throws CharacterCodingException if the sequence of two bytes is malformed and [throwOnMalformed] is true.
*/
private fun codePointFromSurrogate(string: String, high: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (high !in 0xD800..0xDBFF || index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val low = string[index].toInt()
    if (low !in 0xDC00..0xDFFF) {
        return malformed(0, index, throwOnMalformed)
    }
    return 0x10000 + ((high and 0x3FF) shl 10) or (low and 0x3FF)
}

/**
 * Returns code point corresponding to UTF-8 sequence of two bytes,[n] * where the first byte of the sequence is the [byte1] and the second byte is in the [bytes] array at the [index].
* Returns zero if the sequence is malformed and [throwOnMalformed] is false.
* @throws CharacterCodingException if the sequence of two bytes is malformed and [throwOnMalformed] is true.
*/
private fun codePointFrom2(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (byte1 and 0x1E == 0 || index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val byte2 = bytes[index].toInt()
    if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }
    return (byte1 shl 6) xor byte2 xor 0xF80
}

/**
 * Returns code point corresponding to UTF-8 sequence of three bytes,[n] * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].
* Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence
* if the sequence is malformed and [throwOnMalformed] is false.
* @throws CharacterCodingException if the sequence of three bytes is malformed and [throwOnMalformed] is true.
*/
private fun codePointFrom3(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int {
    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }
    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(3, index, throwOnMalformed)
    }
    return (byte1 shl 12) xor (byte2 shl 6) xor (byte3 shl 4) xor (byte4 and 0xF)
}

/**
 * Returns code point corresponding to UTF-8 sequence of four bytes,[n] * where the first byte of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].
* Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence
* if the sequence is malformed and [throwOnMalformed] is false.
* @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true.
*/
    if (index >= endIndex) {
        return malformed(0, index, throwOnMalformed)
    }
    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0x0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }
    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(3, index, throwOnMalformed)
    }
    return (byte1 shl 12) xor (byte2 shl 6) xor (byte3 shl 4) xor byte4 xor 0xF80
}
```
```
malformed(2, index, throwOnMalformed) } in
    return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor
    byte4 xor 0x381F80 // * Maximum number of bytes needed to encode a single char.
    * Code points in `0..0x7F` are encoded in a single byte. * Code points in `0x80..0x7FF` are encoded in two bytes. * Code points in `0x800..0xD7FF` or `0xE000..0xFFFF` are encoded in three bytes. * Surrogate code points in `0xD800..0xDFFF` are not Unicode scalar values, therefore aren't encoded. * Code points in `0x10000..0x10FFFF` are represented by a pair of surrogate `?'s and are encoded in four bytes. * Unprivate const val MAXBYTESPERCHAR = 3 * * * * *
    } The byte sequence a malformed UTF-16 char sequence is replaced by.
    * Unprivate val REPLACEMENTBYTESEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(),
    0xBF.toByte(), 0xBD.toByte()) * * * *
    Encodes the [string] using UTF-8 and returns the resulting [ByteArray].
    * * * * *
    /**
     * Maximum number of bytes needed to encode a single char.
     */
    private const val MAXBYTESPERCHAR = 3

    /**
     * The byte sequence a malformed UTF-16 char sequence is replaced by.
     */
    private val REPLACEMENTBYTESEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(),
    0xBF.toByte(), 0xBD.toByte())

    /**
     * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].
     */
    internal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int,
    throwOnMalformed: Boolean): ByteArray {
        require(startIndex >= 0 && endIndex <= string.length &&
        startIndex <= endIndex)

        val bytes = ByteArray((endIndex - startIndex) * MAXBYTESPERCHAR)
        var byteIndex = 0
        var charIndex = startIndex

        while (charIndex < endIndex) {
            val code = string[charIndex++].toInt()
            when {
                code < 0x80 ->
                    bytes[byteIndex++] = code.toByte()
                code < 0x800 -> {
                    bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()
                    bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                }
                code < 0xD800 || code >= 0xE000 -> {
                    bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()
                    bytes[byteIndex++] = (((code shr 6) and 0x3F) or 0x80).toByte()
                    bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                } else -> { // Surrogate char value
                    val codePoint = codePointFromSurrogate(string, code, charIndex,
                    endIndex, throwOnMalformed)
                    if (codePoint <= 0) {
                        bytes[byteIndex++] = REPLACEMENTBYTESEQUENCE[0]
                        bytes[byteIndex++] = REPLACEMENTBYTESEQUENCE[1]
                        bytes[byteIndex++] = REPLACEMENTBYTESEQUENCE[2]
                    } else {
                        bytes[byteIndex++] = ((codePoint shr 18) or 0xF0).toByte()
                        bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
                        bytes[byteIndex++] = ((codePoint shr 6) and 0x3F) or 0x80).toByte()
                        bytes[byteIndex++] = (codePoint and 0x3F) or 0x80).toByte()
                    }
                }
            }
        }

        return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
    }

    /**
     * The character a malformed UTF-8 byte sequence is replaced by.
     */
    private const val REPLACEMENTCHAR = '\uFFFD'

    /**
     * Decodes the UTF-8 [bytes] array and returns the resulting [String].
     */
    internal fun decodeUtf8(bytes: ByteArray,
    startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {
        require(startIndex >= 0 && endIndex <= bytes.size && startIndex <= endIndex)

        var byteIndex = startIndex
        val stringBuilder = StringBuilder()

        while (byteIndex < endIndex) {
            val byte = bytes[byteIndex++].toInt()
            when {
                byte >= 0 ->
                    stringBuilder.append(byte.toChar())
                byte shr 5 == -2 -> {
                    val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                    if (code <= 0) {
                        stringBuilder.append(REPLACEMENTCHAR) +
                        stringBuilder.append(HEXCHAR) +
                        byteIndex += -code
                    } else {
                        stringBuilder.append(HEXCHAR) +
                        byteIndex += 1
                    }
                } else {
                    val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                    if (code <= 0) {
                        stringBuilder.append(REPLACEMENTCHAR) +
                        stringBuilder.append(HEXCHAR) +
                        byteIndex += -code
                    } else {
                        stringBuilder.append(REPLACEMENTCHAR) +
                        byteIndex += 2
                    }
                } else {
                    val code = codePointFrom4(bytes, byte, byteIndex, endIndex,
                    byteIndex += -code
                } else {
                    val code = codePointFrom4(bytes, byte, byteIndex, endIndex,
                    byteIndex += -code
                }
            }
        }
throwOnMalformed)
if (code <= 0) {
    stringBuilder.append(REPLACEMENT_CHAR)
    byteIndex += -code
} else {
    val high = (code - 0x10000) shr 10 or 0xD800
    val low = (code and 0x3FF) or 0xDC00
    stringBuilder.append(high.toChar())
    stringBuilder.append(low.toChar())
    byteIndex += 3
} else -> {
    malformed(0, byteIndex, throwOnMalformed)
    stringBuilder.append(REPLACEMENT_CHAR)
}
return stringBuilder.toString()
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
public actual enum class DurationUnit(internal val scale: Double) {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS(1e0),
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS(1e3),
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILLISECONDS(1e6),
    /**
     * Time unit representing one second.
     */
    SECONDS(1e9),
    /**
     * Time unit representing one minute.
     */
    MINUTES(60e9),
    /**
     * Time unit representing one hour.
     */
    HOURS(3600e9),
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS(86400e9),
    /**
     * Time unit representing one month, which is always equal to 30 days.
     */
    MONTHS(2592000e9),
    /**
     * Time unit representing one year, which is always equal to 365 days.
     */
    YEARS(3153600000e9);
}

@SinceKotlin("1.3")
@ExperimentalTime
internal actual fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double {
    val sourceCompareTarget = sourceUnit.scale.compareTo(targetUnit.scale)
    return when {
        sourceCompareTarget > 0 -> value * (sourceUnit.scale / targetUnit.scale)
        sourceCompareTarget < 0 -> value / (targetUnit.scale / sourceUnit.scale)
        else -> value
    }
}

/*
 * Copyright 2010-2020 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import org.w3c.performance.GlobalPerformance
import org.w3c.performance.Performance

@SinceKotlin("1.3")
@ExperimentalTime
internal actual object MonotonicTimeSource : TimeSource {
    private val actualSource: TimeSource = run {
        val isNode:
            Boolean = js("typeof process !== 'undefined' && process.versions && !!process.versions.node")
        if
            isNode
            HrTimeSource(js("process").unsafeCast<Process>())
        else
            js("self").unsafeCast<GlobalPerformance?>()?.performance?.let(::PerformanceTimeSource)
                ?: DateNowTimeSource
    }

    override fun markNow(): TimeMark = actualSource.markNow()
}

internal external interface Process {
    fun hrtime(time: Array<Double> = definedExternally): Array<Double>
}

@SinceKotlin("1.3")
@ExperimentalTime
internal class HrTimeSource(val process: Process) : TimeSource {
    override fun markNow(): TimeMark = object : TimeMark() {
        val startedAt = process.hrtime()
        override fun elapsedNow(): Duration = process.hrtime(startedAt).let {
            (seconds, nanos) -> seconds.seconds + nanos.nanoseconds
        }
    }

    override fun toString(): String = "TimeSource(process.hrtime())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal class PerformanceTimeSource(val performance: Performance) : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = performance.now()
    override fun toString(): String = "TimeSource(self.performance.now())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = kotlin.js.Date.now()
    override fun toString(): String = "TimeSource(Date.now())"
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.time

import kotlin.js.json
import kotlin.math.*

internal actual fun formatToExactDecimals(value: Double, decimals: Int): String {
    val rounded = if (decimals == 0) {
        value
    } else {
        val pow = 10.0.pow(decimals)
        kotlin.js.Math.round(abs(value) * pow) / pow * sign(value)
    }
    return rounded.asDynamic().toFixed(decimals).unsafeCast<String>()
}

internal actual fun formatUpToDecimals(value: Double, decimals: Int): String {
    return value.asDynamic().toLocaleString("en-us")
}
json("maximumFractionDigits" to decimals).unsafeCast<String>()

formatScientific(value: Double): String {
    return value.asDynamic().toExponential(2).unsafeCast<String>()
}

internal actual fun formatScientific(value: Double): String {
    return value.asDynamic().toExponential(2).unsafeCast<String>()
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and
 * Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package org.w3c.dom

@Deprecated("Use UnionMessagePortOrWindowProxy instead.", ReplaceWith("UnionMessagePortOrWindowProxy"))
typealias UnionMessagePortOrWindow = UnionMessagePortOrWindowProxy

@Deprecated("Use `as` instead.", ReplaceWith("as"))
var HTMLLinkElement.as_: Any get() = `as` set(value: Any) in

@Deprecated("Use `is` instead.", ReplaceWith("is"))
var ElementCreationOptions.is_: Any get() = `is` set(value: Any)

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!

package org.khronos.webgl

import kotlin.js.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external interface WebGLContextAttributes {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var depth: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var stencil: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var antialias: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var premultipliedAlpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var preserveDrawingBuffer: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var preferLowPowerToHighPerformance: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var failIfMajorPerformanceCaveat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["alpha"] = alpha
    o["depth"] = depth
    o["stencil"] = stencil
    o["antialias"] = antialias
    o["premultipliedAlpha"] = premultipliedAlpha
    o["preserveDrawingBuffer"] = preserveDrawingBuffer
    o["preferLowPowerToHighPerformance"] = preferLowPowerToHighPerformance
    o["failIfMajorPerformanceCaveat"] = failIfMajorPerformanceCaveat
    return o
}

Open Source Used In DNAC Platform Arica  13355

fun viewport(x: Int, y: Int, width: Int, height: Int)

companion object {
    val DEPTH_BUFFER_BIT: Int
    val STENCIL_BUFFER_BIT: Int
    val COLOR_BUFFER_BIT: Int
    val POINTS: Int
    val LINES: Int
    val LINE_LOOP: Int
    val LINE_STRIP: Int
    val TRIANGLES: Int
    val TRIANGLE_STRIP: Int
    val TRIANGLE_FAN: Int

    val SRC_COLOR: Int
    val ONE_MINUS_SRC_COLOR: Int
    val SRC_ALPHA: Int
    val ONE_MINUS_SRC_ALPHA: Int
    val DST_ALPHA: Int
    val ONE_MINUS_DST_ALPHA: Int
    val DST_COLOR: Int
    val ONE_MINUS_DST_COLOR: Int
    val SRC_ALPHA_SATURATE: Int
    val BLEND_EQUATION: Int
    val BLEND_EQUATION_RGB: Int
    val BLEND_EQUATION_ALPHA: Int

    val ARRAY_BUFFER: Int
    val ELEMENT_ARRAY_BUFFER: Int
    val ARRAY_BUFFER_BINDING: Int
    val ELEMENT_ARRAY_BUFFER_BINDING: Int
    val DYNAMIC_DRAW: Int
    val BUFFER_SIZE: Int
    val BUFFER_USAGE: Int
    val CURRENT_VERTEX_ATTRIB: Int
    val FRONT: Int
    val BACK: Int
    val FRONT_AND_BACK: Int
    val CULL_FACE: Int
    val BLEND: Int
    val DITHER: Int
    val STENCIL_TEST: Int
    val DEPTH_TEST: Int
    val SCISSOR_TEST: Int
    val POLYGON_OFFSET_FILL: Int
    val SAMPLE_BUFFERS: Int
    val SAMPLES: Int
    val COMPRESSED_TEXTURE_FORMATS: Int
    val DONT_CARE: Int
    val FASTEST: Int
    val NICEST: Int
    val GENERATE_MIPMAP_HINT: Int
    val BYTE: Int
    val UNSIGNED_BYTE: Int
    val SHORT: Int
    val UNSIGNED_SHORT: Int
    val INT: Int
    val UNSIGNED_INT: Int
    val FLOAT: Int
    val DEPTH_COMPONENT: Int
    val ALPHA: Int
    val RGB: Int
    val RGBA: Int

UNSIGNED_SHORT_4_4_4_4: Int
UNSIGNED_SHORT_5_6_5: Int
UNSIGNED_SHORT_5_5_5_1: Int
UNSIGNED_SHORT_5_5_5_1: Int
MAX_VERTEX_ATTRIBS: Int
MAX_VERTEX_UNIFORM_VECTORS: Int
MAX_VARYING_VECTORS: Int
MAX_COMBINED_TEXTURE_IMAGE_UNITS: Int
MAX_VERTEX_TEXTURE_IMAGE_UNITS: Int
MAX_TEXTURE_IMAGE_UNITS: Int
MAX_FRAGMENT_UNIFORM_VECTORS: Int
SHADER_TYPE: Int
DELETE_STATUS: Int
LINK_STATUS: Int
VALIDATE_STATUS: Int
ACTIVE_UNIFORMS: Int
ACTIVE_ATTRIBUTES: Int
SHADING_LANGUAGE_VERSION: Int
CURRENT_PROGRAM: Int
NEVER: Int
LESS: Int
EQUAL: Int
LEQUAL: Int
GREATER: Int
NOTEQUAL: Int
GEQUAL: Int
ALWAYS: Int
KEEP: Int
REPLACE: Int
INCR: Int
INCR_WRAP: Int
DECR: Int
DECR_WRAP: Int
VENDOR: Int
RENDERER: Int
VERSION: Int
NEAREST: Int
LINEAR: Int
NEAREST_MIPMAP_NEAREST: Int
LINEAR_MIPMAP_NEAREST: Int
TEXTURE_MAG_FILTER: Int
TEXTURE_MIN_FILTER: Int
TEXTURE_WRAP_S: Int
TEXTURE_WRAP_T: Int
TEXTURE_2D: Int
TEXTURE: Int
TEXTURE_CUBE_MAP: Int
TEXTURE_BINDING_CUBE_MAP: Int
TEXTURE_CUBE_MAP_POSITIVE_X: Int
TEXTURE_CUBE_MAP_NEGATIVE_X: Int
TEXTURE_CUBE_MAP_POSITIVE_Y: Int
TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
TEXTURE_CUBE_MAP_POSITIVE_Z: Int
TEXTURE_CUBE_MAP_NEGATIVE_Z: Int
MAX_CUBE_MAP_TEXTURE_SIZE: Int
TEXTURE0: Int
TEXTURE1: Int
TEXTURE2: Int
TEXTURE3: Int
TEXTURE4: Int
TEXTURE5: Int
TEXTURE6: Int
TEXTURE7: Int
TEXTURE8: Int
TEXTURE9: Int
TEXTURE10: Int
TEXTURE11: Int
TEXTURE12: Int
TEXTURE13: Int
TEXTURE14: Int
TEXTURE15: Int
TEXTURE16: Int
TEXTURE17: Int
TEXTURE18: Int
TEXTURE19: Int
TEXTURE20: Int
TEXTURE21: Int
TEXTURE22: Int
TEXTURE23: Int
TEXTURE24: Int
TEXTURE25: Int
TEXTURE26: Int
TEXTURE27: Int
TEXTURE28: Int
TEXTURE29: Int
TEXTURE30: Int
TEXTURE31: Int
ACTIVE_TEXTURE: Int
REPEAT: Int
CLAMP_TO_EDGE: Int
MIRRORED_REPEAT: Int
FLOAT_VEC2: Int
FLOAT_VEC3: Int
FLOAT_VEC4: Int
INT_VEC2: Int
INT_VEC3: Int
INT_VEC4: Int
BOOL: Int
BOOL_VEC2: Int
BOOL_VEC3: Int
BOOL_VEC4: Int
FLOAT_MAT2: Int
FLOAT_MAT3: Int
FLOAT_MAT4: Int
SAMPLER_2D: Int
SAMPLER_CUBE: Int
VERTEX_ATTRIB_ARRAY_ENABLED: Int
VERTEX_ATTRIB_ARRAY_SIZE: Int
VERTEX_ATTRIB_ARRAY_STRIDE: Int
VERTEX_ATTRIB_ARRAY_TYPE: Int
VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
VERTEX_ATTRIB_ARRAY_POINTER: Int
VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
IMPLEMENTATION_COLOR_READ_TYPE: Int
IMPLEMENTATION_COLOR_READ_FORMAT: Int
COMPILE_STATUS: Int
LOW_FLOAT: Int
MEDIUM_FLOAT: Int
HIGH_FLOAT: Int
LOW_INT: Int
MEDIUM_INT: Int
HIGH_INT: Int
FRAMEBUFFER: Int
RENDERBUFFER: Int
RGBA4: Int
RGB5_A1: Int
RGB565: Int
DEPTH_COMPONENT16: Int
STENCIL_INDEX: Int
STENCIL_INDEX8: Int
DEPTH_STENCIL: Int
RENDERBUFFER_WIDTH: Int
RENDERBUFFER_HEIGHT: Int
RENDERBUFFER_INTERNAL_FORMAT: Int
RENDERBUFFER_RED_SIZE: Int
RENDERBUFFER_GREEN_SIZE: Int
RENDERBUFFER_BLUE_SIZE: Int
RENDERBUFFER_ALPHA_SIZE: Int
RENDERBUFFER_DEPTH_SIZE: Int
RENDERBUFFER_STENCIL_SIZE: Int
SHADER_TYPE: Int
RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_BINDING: Int
val RENDERBUFFER_BINDING: Int
val MAX_RENDERBUFFER_SIZE: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
val BROWSER_DEFAULT_WEBGL: Int
val DEPTH_BUFFER_BIT: Int
val STENCIL_BUFFER_BIT: Int
val COLOR_BUFFER_BIT: Int
val POINTS: Int
val LINES: Int
val LINE_LOOP: Int
val LINE_STRIP: Int
val TRIANGLES: Int
val TRIANGLE_STRIP: Int
val TRIANGLE_FAN: Int
val ZERO: Int
val ONE: Int
val SRC_COLOR: Int
val ONE_MINUS_SRC_COLOR: Int
val SRC_ALPHA: Int
val ONE_MINUS_SRC_ALPHA: Int
val DST_ALPHA: Int
val ONE_MINUS_DST_ALPHA: Int
val SRC_ALPHA_SATURATE: Int
val FUNC_ADD: Int
val BLEND_EQUATION: Int
val BLEND_EQUATION_RGB: Int
val BLEND_EQUATION_ALPHA: Int
val FUNC_SUBTRACT: Int
val BLEND_DST_RGB: Int
val BLEND_SRC_RGB: Int
val BLEND_DST_ALPHA: Int
val BLEND_SRC_ALPHA: Int
val CONSTANT_COLOR: Int
val ONE_MINUS_CONSTANT_COLOR: Int
val CONSTANT_ALPHA: Int
val ONE_MINUS_CONSTANT_ALPHA: Int
val BLEND_COLOR: Int
val ARRAY_BUFFER: Int
val ELEMENT_ARRAY_BUFFER: Int
val ARRAY_BUFFER_BINDING: Int
val ELEMENT_ARRAY_BUFFER_BINDING: Int
val STREAM_DRAW: Int
val STATIC_DRAW: Int
val DYNAMIC_DRAW: Int
val BUFFER_SIZE: Int
val BUFFER_USAGE: Int
val CURRENT_VERTEX_ATTRIB: Int
val FRONT: Int
val BACK: Int
val FRONT_AND_BACK: Int
val CULL_FACE: Int
val BLEND: Int
val DITHER: Int
val STENCIL_TEST: Int
val DEPTH_TEST: Int
val SCISSOR_TEST: Int
val POLYGON_OFFSET_FILL: Int
val SAMPLE_ALPHA_TO_COVERAGE: Int
val SAMPLE_COVERAGE: Int
val NO_ERROR: Int
val INVALID_ENUM: Int
val INVALID_VALUE: Int
val INVALID_OPERATION: Int
val OUT_OF_MEMORY: Int
val CW: Int
val CCW: Int
val LINE_WIDTH: Int
val ALIASED_POINT_SIZE_RANGE: Int
val ALIASED_LINE_WIDTH_RANGE: Int
val CULL_FACE_MODE: Int
val FRONT_FACE: Int
val DEPTH_RANGE: Int
val DEPTH_WIDTH: Int
val DEPTH_CLEAR_VALUE: Int
val DEPTH_FUNC: Int
val STENCIL_CLEAR_VALUE: Int
val STENCIL_FUNC: Int
val STENCIL_FAIL: Int
val STENCIL_PASS_DEPTH_FAIL: Int
val STENCIL_PASS_DEPTH_PASS: Int
val STENCIL_REF: Int
val STENCIL_VALUE_MASK: Int
val STENCIL_WRITE_MASK: Int
val STENCIL_BACK_FUNC: Int
val STENCIL_BACK_FAIL: Int
val STENCIL_BACK_PASS_DEPTH_FAIL: Int
val STENCIL_BACK_PASS_DEPTH_PASS: Int
val STENCIL_BACK_REF: Int
val STENCIL_BACK_VALUE_MASK: Int
val STENCIL_BACK_WRITE_MASK: Int
val VIEWPORT: Int
val SCISSOR_BOX: Int
val COLOR_CLEAR_VALUE: Int
val COLOR_WRITEMASK: Int
val UNPACK_ALIGNMENT: Int

/**
 * Exposes the JavaScript [WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to Kotlin
 */

public external abstract class WebGLRenderingContext : WebGLRenderingContextBase, RenderingContext {
companion object {
    val DEPTH_BUFFER_BIT: Int
    val STENCIL_BUFFER_BIT: Int
    val COLOR_BUFFER_BIT: Int
    val POINTS: Int
    val LINES: Int
    val LINE_LOOP: Int
    val LINE_STRIP: Int
    val TRIANGLES: Int
    val TRIANGLE_STRIP: Int
    val TRIANGLE_FAN: Int
    val ZERO: Int
    val ONE: Int
    val SRC_COLOR: Int
    val ONE_MINUS_SRC_COLOR: Int
    val SRC_ALPHA: Int
    val ONE_MINUS_SRC_ALPHA: Int
    val DST_ALPHA: Int
    val ONE_MINUS_DST_ALPHA: Int
    val SRC_ALPHA_SATURATE: Int
    val FUNC_ADD: Int
    val BLEND_EQUATION: Int
    val BLEND_EQUATION_RGB: Int
    val BLEND_EQUATION_ALPHA: Int
    val FUNC_SUBTRACT: Int
    val BLEND_DST_RGB: Int
    val BLEND_SRC_RGB: Int
    val BLEND_DST_ALPHA: Int
    val BLEND_SRC_ALPHA: Int
    val CONSTANT_COLOR: Int
    val ONE_MINUS_CONSTANT_COLOR: Int
    val CONSTANT_ALPHA: Int
    val ONE_MINUS_CONSTANT_ALPHA: Int
    val BLEND_COLOR: Int
    val ARRAY_BUFFER: Int
    val ELEMENT_ARRAY_BUFFER: Int
    val ARRAY_BUFFER_BINDING: Int
    val ELEMENT_ARRAY_BUFFER_BINDING: Int
    val STREAM_DRAW: Int
    val STATIC_DRAW: Int
    val DYNAMIC_DRAW: Int
    val BUFFER_SIZE: Int
    val BUFFER_USAGE: Int
    val CURRENT_VERTEX_ATTRIB: Int
    val FRONT: Int
    val BACK: Int
    val FRONT_AND_BACK: Int
    val CULL_FACE: Int
    val BLEND: Int
    val DITHER: Int
    val STENCIL_TEST: Int
    val DEPTH_TEST: Int
    val SCISSOR_TEST: Int
    val POLYGON_OFFSET_FILL: Int
    val SAMPLE_ALPHA_TO_COVERAGE: Int
    val SAMPLE_COVERAGE: Int
    val NO_ERROR: Int
    val INVALID_ENUM: Int
    val INVALID_VALUE: Int
    val INVALID_OPERATION: Int
    val OUT_OF_MEMORY: Int
    val CW: Int
    val CCW: Int
    val LINE_WIDTH: Int
    val ALIASED_POINT_SIZE_RANGE: Int
    val ALIASED_LINE_WIDTH_RANGE: Int
    val CULL_FACE_MODE: Int
    val FRONT_FACE: Int
    val DEPTH_RANGE: Int
    val DEPTH_WIDTH: Int
    val DEPTH_CLEAR_VALUE: Int
    val DEPTH_FUNC: Int
    val STENCIL_CLEAR_VALUE: Int
    val STENCIL_FUNC: Int
    val STENCIL_FAIL: Int
    val STENCIL_PASS_DEPTH_FAIL: Int
    val STENCIL_PASS_DEPTH_PASS: Int
    val STENCIL_REF: Int
    val STENCIL_VALUE_MASK: Int
    val STENCIL_WRITE_MASK: Int
    val STENCIL_BACK_FUNC: Int
    val STENCIL_BACK_FAIL: Int
    val STENCIL_BACK_PASS_DEPTH_FAIL: Int
    val STENCIL_BACK_PASS_DEPTH_PASS: Int
    val STENCIL_BACK_REF: Int
    val STENCIL_BACK_VALUE_MASK: Int
    val STENCIL_BACK_WRITE_MASK: Int
    val VIEWPORT: Int
    val SCISSOR_BOX: Int
    val COLOR_CLEAR_VALUE: Int
    val COLOR_WRITEMASK: Int
    val UNPACK_ALIGNMENT: Int
}
val PACK_ALIGNMENT: Int = MAX_TEXTURE_SIZE: Int = MAX_VERTEX_ATTRIBS: Int
val POLYGON_OFFSET_UNITS: Int = POLYGON_OFFSET_FACTOR: Int
val TEXTURE_BINDING_2D: Int = SAMPLE_BUFFERS: Int = SAMPLES: Int
val SAMPLE_COVERAGE_VALUE: Int = SAMPLE_COVERAGE_INVERT: Int
val COMPRESSED_TEXTURE_FORMATS: Int = DONT_CARE: Int = FASTEST: Int
val NICEST: Int = GENERATE_MIPMAP_HINT: Int = BYTE: Int = UNSIGNED_BYTE: Int
val SHORT: Int = UNSIGNED_SHORT: Int = INT: Int = UNSIGNED_INT: Int
val LUMINANCE_ALPHA: Int = UNSIGNED_SHORT_4_4_4_4: Int = UNSIGNED_SHORT_5_5_5_1: Int
val UNSIGNED_SHORT_5_6_5: Int = FRAGMENT_SHADER: Int = VERTEX_SHADER: Int
val MAX_VERTEX_ATTRIBS: Int = MAX_VERTEX_UNIFORM_VECTORS: Int = MAX_VARYING_VECTORS: Int
val MAX_VERTEX_TEXTURE_IMAGE_UNITS: Int = MAX_TEXTURE_IMAGE_UNITS: Int
val MAX_FRAGMENT_UNIFORM_VECTORS: Int = SHADER_TYPE: Int
val GENERATE_MIPMAP_HINT: Int = DELETE_STATUS: Int
val VALIDATE_STATUS: Int
val SHADING_LANGUAGE_VERSION: Int = CURRENT_PROGRAM: Int
val TEXCOORD0: Int = TEXCOORD1: Int = TEXCOORD2: Int
val TEXCOORD3: Int = TEXCOORD4: Int = TEXCOORD5: Int
val TEXCOORD6: Int = TEXCOORD7: Int = TEXCOORD8: Int = TEXCOORD9: Int
val TEXCOORD14: Int = TEXCOORD15: Int = TEXCOORD16: Int = TEXCOORD17: Int
val TEXCOORD18: Int = TEXCOORD19: Int = TEXCOORD20: Int = TEXCOORD21: Int
val TEXCOORD30: Int = TEXCOORD31: Int
val TEST_PASS: Int = CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int = FLOAT_VEC3: Int = FLOAT_VEC4: Int
val INT_VEC2: Int = INT_VEC3: Int = INT_VEC4: Int
val FLOAT_MAT2: Int = FLOAT_MAT3: Int = FLOAT_MAT4: Int
val SAMPLER_2D: Int = SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val COMPILE_STATUS: Int
val LOW_FLOAT: Int
val MEDIUM_FLOAT: Int
val HIGH_FLOAT: Int
val LOW_INT: Int
val MEDIUM_INT: Int
val HIGH_INT: Int
val FRAMEBUFFER: Int
val RENDERBUFFER: Int
val RGBA4: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT16: Int
val STENCIL_INDEX: Int
val STENCIL_INDEX8: Int
val DEPTH_STENCIL: Int
val RENDERBUFFER_WIDTH: Int
val RENDERBUFFER_HEIGHT: Int
val RENDERBUFFER_INTERNAL_FORMAT: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_BLUE_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_BINDING: Int
val RENDERBUFFER_BINDING: Int
val MAX_RENDERBUFFER_SIZE: Int
val INVALID_FRAMEBUFFER_OPERATION: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
val BROWSER_DEFAULT_WEBGL: Int

/**
 * Exposes the JavaScript [WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin
 */
public external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit = definedExternally) : Event {
    open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WebGLContextEventInit : EventInit {
    var statusMessage: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["statusMessage"] = statusMessage
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin
 */
public external open class ArrayBuffer(length: Int) : BufferDataSource {
    open val byteLength: Int
    fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer

    companion object {
        fun isView(value: Any?): Boolean
    }
}

/**
 * Exposes the JavaScript [Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin
 */
public external open class Int8Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Int8Array)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Int8Array, offset: Int = definedExternally)
    fun set(array: ArrayBufferView, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int8Array
}

open class Open Source Used In DNAC Platform Arica 13361
object { ```
    val BYTES_PER_ELEMENT: Int
```
}

@kotlin.internal.InlineOnly
public inline operator fun Int8Array.get(index: Int): Byte = asDynamic()[index]@kotlin.internal.InlineOnly
public inline operator fun Int8Array.set(index: Int, value: Byte) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```
}

@kotlin.internal.InlineOnly
public inline operator fun Int8Array(array: Byte, offset: Int) = ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

/**
 * Exposes the JavaScript [Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array) to Kotlin
 */
public external open class Uint8Array : ArrayBufferView { ```
    constructor(length: Int)
    constructor(array: Uint8Array)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    fun set(array: Uint8Array, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint8Array

    companion object { ```
        val BYTES_PER_ELEMENT: Int
```
}
```

@kotlin.internal.InlineOnly
public inline operator fun Uint8Array.get(index: Int): Byte = ```
    asDynamic()[index]@kotlin.internal.InlineOnly
```

@kotlin.internal.InlineOnly
public inline operator fun Uint8Array.set(index: Int, value: Byte) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

/**
 * Exposes the JavaScript [Uint8ClampedArray](https://developer.mozilla.org/en/docs/Web/API/Uint8ClampedArray) to Kotlin
 */
public external open class Uint8ClampedArray : ArrayBufferView { ```
    constructor(length: Int)
    constructor(array: Uint8ClampedArray)
    constructor(array: Array<Byte>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    fun set(array: Uint8ClampedArray, offset: Int = definedExternally)
    fun set(array: Array<Byte>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint8ClampedArray

    companion object { ```
        val BYTES_PER_ELEMENT: Int
```
}
```

@kotlin.internal.InlineOnly
public inline operator fun Uint8ClampedArray.get(index: Int): Byte = ```
    asDynamic()[index]@kotlin.internal.InlineOnly
```

@kotlin.internal.InlineOnly
public inline operator fun Uint8ClampedArray.set(index: Int, value: Byte) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

/**
 * Exposes the JavaScript [Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin
 */
public external open class Int16Array : ArrayBufferView { ```
    constructor(length: Int)
    constructor(array: Int16Array)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    fun set(array: Int16Array, offset: Int = definedExternally)
    fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int16Array

    companion object { ```
        val BYTES_PER_ELEMENT: Int
```
}
```

@kotlin.internal.InlineOnly
public inline operator fun Int16Array.get(index: Int): Short = ```
    asDynamic()[index]@kotlin.internal.InlineOnly
```

@kotlin.internal.InlineOnly
public inline operator fun Int16Array.set(index: Int, value: Short) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

/**
 * Exposes the JavaScript [Uint16Array](https://developer.mozilla.org/en/docs/Web/API/Uint16Array) to Kotlin
 */
public external open class Uint16Array : ArrayBufferView { ```
    constructor(length: Int)
    constructor(array: Uint16Array)
    constructor(array: Array<Short>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    fun set(array: Uint16Array, offset: Int = definedExternally)
    fun set(array: Array<Short>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint16Array

    companion object { ```
        val BYTES_PER_ELEMENT: Int
```
}
```

@kotlin.internal.InlineOnly
public inline operator fun Uint16Array.get(index: Int): Short = ```
    asDynamic()[index]@kotlin.internal.InlineOnly
```

@kotlin.internal.InlineOnly
public inline operator fun Uint16Array.set(index: Int, value: Short) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

/**
 * Exposes the JavaScript [Int32Array](https://developer.mozilla.org/en/docs/Web/API/Int32Array) to Kotlin
 */
public external open class Int32Array : ArrayBufferView { ```
    constructor(length: Int)
    constructor(array: Int32Array)
    constructor(array: Array<Int>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    fun set(array: Int32Array, offset: Int = definedExternally)
    fun set(array: Array<Int>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Int32Array

    companion object { ```
        val BYTES_PER_ELEMENT: Int
```
}
```

@kotlin.internal.InlineOnly
public inline operator fun Int32Array.get(index: Int): Int = ```
    asDynamic()[index]@kotlin.internal.InlineOnly
```

@kotlin.internal.InlineOnly
public inline operator fun Int32Array.set(index: Int, value: Int) { ```
    [asDynamic()][index] = value @kotlin.internal.InlineOnly
```

Open Source Used In DNAC Platform Arica 13362
Int32Array.set(index: Int, value: Int) { asDynamic() }[index] = value

Uint32Array.set(index: Int, value: Int) { asDynamic() }[index] = value

/**
 * Exposes the JavaScript [Uint32Array](https://developer.mozilla.org/en/docs/Web/API/Uint32Array) to Kotlin
 */
public external open class Uint32Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Uint32Array)
    constructor(array: Array<Int>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Uint32Array, offset: Int = definedExternally)
    fun set(array: Array<Int>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Uint32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.get(index: Int): Int = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Uint32Array.set(index: Int, value: Int) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin
 */
public external open class Float32Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float32Array)
    constructor(array: Array<Float>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float32Array, offset: Int = definedExternally)
    fun set(array: Array<Float>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float32Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.get(index: Int): Float = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float32Array.set(index: Int, value: Float) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin
 */
public external open class Float64Array : ArrayBufferView {
    constructor(length: Int)
    constructor(array: Float64Array)
    constructor(array: Array<Double>)
    constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
    open val length: Int
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun set(array: Float64Array, offset: Int = definedExternally)
    fun set(array: Array<Double>, offset: Int = definedExternally)
    fun subarray(start: Int, end: Int): Float64Array

    companion object {
        val BYTES_PER_ELEMENT: Int
    }
}

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.get(index: Int): Double = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun Float64Array.set(index: Int, value: Double) { asDynamic()[index] = value }

/**
 * Exposes the JavaScript [DataView](https://developer.mozilla.org/en/docs/Web/API/DataView) to Kotlin
 */
public external open class DataView(buffer: ArrayBuffer, byteOffset: Int = definedExternally, byteLength: Int = definedExternally) : ArrayBufferView {
    override val buffer: ArrayBuffer
    override val byteOffset: Int
    override val byteLength: Int
    fun getInt8(byteOffset: Int): Byte
    fun getUint8(byteOffset: Int): Byte
    fun getInt16(byteOffset: Int, littleEndian: Boolean = definedExternally): Short
    fun getUint16(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getInt32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getUint32(byteOffset: Int, littleEndian: Boolean = definedExternally): Int
    fun getFloat32(byteOffset: Int, littleEndian: Boolean = definedExternally): Float
    fun getFloat64(byteOffset: Int, littleEndian: Boolean = definedExternally): Double
    fun setInt8(byteOffset: Int, value: Byte)
    fun setUint8(byteOffset: Int, value: Byte)
    fun setInt16(byteOffset: Int, value: Short, littleEndian: Boolean = definedExternally)
    fun setUint16(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setInt32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setUint32(byteOffset: Int, value: Int, littleEndian: Boolean = definedExternally)
    fun setFloat32(byteOffset: Int, value: Float, littleEndian: Boolean = definedExternally)
    fun setFloat64(byteOffset: Int, value: Double, littleEndian: Boolean = definedExternally)
}

public external interface BufferDataSource

public external interface TexImageSource /*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details

Open Source Used In DNAC Platform Arica 13363
Open Source Used In DNAC Platform Arica 13364

```java
public external interface ClipboardEventInit : EventInit {
    var clipboardData: DataTransfer? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["clipboardData"] = clipboardData
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 *[ClipboardEvent](https://developer.mozilla.org/en/docs/Web/API/ClipboardEvent) to Kotlin
 */
public external open class ClipboardEvent(type: String, eventInitDict: ClipboardEventInit = definedExternally) : Event {
    open
    val clipboardData: DataTransfer?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

/**
 * Exposes the JavaScript
 *[Clipboard](https://developer.mozilla.org/en/docs/Web/API/Clipboard) to Kotlin
 */
public external abstract class Clipboard : EventTarget {
    fun read(): Promise<DataTransfer>
    fun readText(): Promise<String>
    fun write(data: DataTransfer): Promise<Unit>
    fun writeText(data: String): Promise<Unit>
}

public external interface ClipboardPermissionDescriptor {
    var allowWithoutGesture: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ClipboardPermissionDescriptor(allowWithoutGesture: Boolean? = false): ClipboardPermissionDescriptor {
    val o = js("({})")
    o["allowWithoutGesture"] = allowWithoutGesture
    return o
}
```

---

[/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTOMATICALLY GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.dom.css

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external abstract class MediaList : ItemArrayLike<String> {
    open var mediaText: String
    fun appendMedium(medium: String)
    fun deleteMedium(medium: String)
    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun MediaList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript
 *[StyleSheet](https://developer.mozilla.org/en/docs/Web/API/StyleSheet) to Kotlin
 */
public external abstract class StyleSheet {
    open val type: String
    open val href: String?
    open val ownerNode: UnionElementOrProcessingInstruction?
    open val parentStyleSheet: StyleSheet?
    open val title: String?
    open var disabled: Boolean
}

/**
 * Exposes the JavaScript
 *[CSSStyleSheet](https://developer.mozilla.org/en/docs/Web/API/CSSStyleSheet) to Kotlin
 */
public external abstract class CSSStyleSheet : StyleSheet {
    open val ownerRule: CSSRule?
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)
}

/**
 * Exposes the JavaScript
 *[StyleSheetList](https://developer.mozilla.org/en/docs/Web/API/StyleSheetList) to Kotlin
 */
public external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {
    override fun item(index: Int): StyleSheet?
}

@kotlin.internal.InlineOnly
public inline operator fun StyleSheetList.get(index: Int): StyleSheet? = asDynamic()[index]

/**
 * Exposes the JavaScript
 *[LinkStyle](https://developer.mozilla.org/en/docs/Web/API/LinkStyle) to Kotlin
 */
public external interface LinkStyle {
    val sheet: StyleSheet?
        get() = definedExternally
}

/**
 * Exposes the JavaScript
 *[CSSRuleList](https://developer.mozilla.org/en/docs/Web/API/CSSRuleList) to Kotlin
 */
public external abstract class CSSRuleList : ItemArrayLike<CSSRule> {
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)
}
val o = js(`({})`)\n       o["view"] = view\n       o["detail"] = detail\n       o["bubbles"] = bubbles\n       o["cancelable"] = cancelable\n       o["composed"] = composed\n       return o\n\n* Exposes the JavaScript

[FocusEvent](https://developer.mozilla.org/en/docs/Web/API/FocusEvent) to Kotlin

```kotlin
/**
 * Exposes the JavaScript

[FocusEvent](https://developer.mozilla.org/en/docs/Web/API/FocusEvent) to Kotlin
 */
public external open class FocusEvent(type: String, eventInitDict: FocusEventInit = definedExternally) : UIEvent {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

public external interface FocusEventInit : UIEventInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js(`({})`)\n    o["relatedTarget"] = relatedTarget\n    o["view"] = view\n    o["detail"] = detail\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n\n* Exposes the JavaScript

[MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin

```kotlin
/**
 * Exposes the JavaScript

[MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin
 */
public external open class MouseEvent(type: String, eventInitDict: MouseEventInit = definedExternally) : UIEvent, UnionElementOrMouseEvent {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

public external interface MouseEventInit : EventModifierInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js(`({})`)\n    o["relatedTarget"] = relatedTarget\n    o["view"] = view\n    o["detail"] = detail\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n\n* Exposes the JavaScript

[MouseEvent](https://developer.mozilla.org/en/docs/Web/API/MouseEvent) to Kotlin

```kotlin
/**
 * Exposes the JavaScript

[MouseMove](https://developer.mozilla.org/en/docs/Web/API/MouseMove) to Kotlin
 */
public external class MoveEvent(type: String, eventInitDict: MoveEventInit = definedExternally) : UIEvent, MoveEventInit {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

public external interface MoveEventInit : EventModifierInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js(`({})`)\n    o["relatedTarget"] = relatedTarget\n    o["view"] = view\n    o["detail"] = detail\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n\n* Exposes the JavaScript

[MouseMove](https://developer.mozilla.org/en/docs/Web/API/MouseMove) to Kotlin

```kotlin
/**
 * Exposes the JavaScript

[MouseWheel](https://developer.mozilla.org/en/docs/Web/API/MouseWheel) to Kotlin
 */
public external class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : UIEvent, Event, WheelEventInit {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

public external interface WheelEventInit : EventModifierInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js(`({})`)\n    o["relatedTarget"] = relatedTarget\n    o["view"] = view\n    o["detail"] = detail\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n\n* Exposes the JavaScript

[MouseWheel](https://developer.mozilla.org/en/docs/Web/API/MouseWheel) to Kotlin
false */
    get() = definedExternally
    set(value) = definedExternally
    var altKey: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var metaKey: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierAltGraph: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierCapsLock: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierFn: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierFnLock: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierHyper: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierNumLock: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierScrollLock: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierSuper: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierSymbol: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var modifierSymbolLock: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 [WheelEvent](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin
 */
public external open class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {
    open val deltaX: Double
    open val deltaY: Double
    open val deltaZ: Double
    open val deltaMode: Int

    companion object {
        val DOM_DELTA_PIXEL: Int
        val DOM_DELTA_LINE: Int
        val DOM_DELTA_PAGE: Int
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface WheelEventInit : MouseEventInit {
    var deltaX: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaY: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaZ: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaMode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["deltaX"] = deltaX
    o["deltaY"] = deltaY
    o["deltaZ"] = deltaZ
    o["deltaMode"] = deltaMode
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
}
```javascript
/**
 * Exposes the JavaScript [InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to Kotlin
 */
public external open class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) :
    UIEvent {
    open val data: String
    open val isComposing: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface InputEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["isComposing"] = isComposing
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin
 */
public external open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {
    open val key: String
    open val code: String
    open val location: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean
    open val repeat: Boolean

    open val isComposing: Boolean
    open val charCode: Int
    open val keyCode: Int
    open val which: Int

    fun getModifierState(keyArg: String): Boolean

    companion object {
        val DOM_KEY_LOCATION_STANDARD: Int
        val DOM_KEY_LOCATION_LEFT: Int
        val DOM_KEY_LOCATION_RIGHT: Int
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface KeyboardEventInit : EventModifierInit {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var code: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var location: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var repeat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["key"] = key
    o["code"] = code
    o["location"] = location
    o["repeat"] = repeat
    o["isComposing"] = isComposing
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
```
The given text appears to be a code snippet from a documentation or source code file. It contains JavaScript and Kotlin code, along with comments explaining the functionalities and interfaces exposed to Kotlin. The code includes the definition and usage of various events, event targets, and listeners, as well as class definitions for collections and interfaces related to Web API functionalities.

For example, there is a class definition for `CompositionEvent` that exposes the JavaScript `CompositionEvent` to Kotlin, and similar definitions for other events like `Event`, `EventTarget`, and `EventListener`. These classes are used to interact with the browser's event system, allowing Kotlin applications to respond to user interactions.

The comments indicate that this file is auto-generated, as noted by the message at the bottom: `// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!`.

The code also includes imports for various Web APIs, such as `org.khronos.webgl.*`, `org.w3c.dom.*`, and Java-based classes like `String`, `Boolean`, and `Array`. These imports are used to support the JavaScript and Web API functionalities exposed in the Kotlin code.

Overall, the document contains a comprehensive and detailed exposition of how to use various JavaScript and Web API functionalities within a Kotlin environment, ensuring seamless integration between the two languages.
String

/**
 * Exposes the JavaScript [HTMLOptionsCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionsCollection) to Kotlin
 */
public external abstract class HTMLOptionsCollection : HTMLCollection {
    override var length: Int
    open
    var selectedIndex: Int
    fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
    fun remove(index: Int)
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLOptionsCollection.set(index: Int, option: HTMLOptionElement?) { asDynamic()[index] = option }

/**
 * Exposes the JavaScript [HTMLElement](https://developer.mozilla.org/en/docs/Web/API/HTMLElement) to Kotlin
 */
public external abstract class HTMLElement : Element, GlobalEventHandlers, DocumentAndElementEventHandlers, ElementContentEditable, ElementCSSInlineStyle {
    open var title: String
    open var lang: String
    open var translate: Boolean
    open var dir: String
    open val dataset: DOMStringMap
    open var hidden: Boolean
    open var tabIndex: Int
    open var accessKey: String
    open val accessKeyLabel: String
    open var draggable: Boolean
    open val dropzone: DOMTokenList
    open var contextMenu: HTMLMenuElement
    open var spellcheck: Boolean
    open var innerText: String
    open val offsetParent: Element
    open val offsetTop: Int
    open val offsetLeft: Int
    open val offsetWidth: Int
    open val offsetHeight: Int
    fun click()
    fun focus()
    fun blur()
    fun forceSpellCheck()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
    }
}

/**
 * Exposes the JavaScript [DOMStringMap](https://developer.mozilla.org/en/docs/Web/API/DOMStringMap) to Kotlin
 */
public external abstract class DOMStringMap

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.get(name: String): String? = asDynamic()[name]

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.set(name: String, value: String) { asDynamic()[name] = value }

/**
 * Exposes the JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin
 */
public external abstract class HTMLHtmlElement : HTMLElement {
    open var version: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
    }
}
val DOCUMENT_POSITION_FOLLOWING: Short = val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short = val

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short = Exposes the JavaScript
[HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin

*public external abstract class HTMLHeadElement : HTMLElement {

    companion object {

        val ELEMENT_NODE: Short = val ATTRIBUTE_NODE: Short = val
        val TEXT_NODE: Short = val
        val CDATA_SECTION_NODE: Short = val ENTITY_REFERENCE_NODE: Short = val
        val ENTITY_NODE: Short = val
        val PROCESSING_INSTRUCTION_NODE: Short = val
        val COMMENT_NODE: Short = val
        val DOCUMENT_NODE: Short = val
        val DOCUMENT_TYPE_NODE: Short = val
        val DOCUMENT_FRAGMENT_NODE: Short = val
        val NOTATION_NODE: Short = val
        val DOCUMENT_POSITION_DISCONNECTED: Short = val
        val DOCUMENT_POSITION_PRECEDING: Short = val
        val DOCUMENT_POSITION_FOLLOWING: Short = val
        val DOCUMENT_POSITION_CONTAINS: Short = val
        val DOCUMENT_POSITION_CONTAINED_BY: Short = val
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short = Exposes the JavaScript
[HTMLTitleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTitleElement) to Kotlin

*public external abstract class HTMLTitleElement : HTMLElement {

    companion object {

        val ELEMENT_NODE: Short = val ATTRIBUTE_NODE: Short = val
        val TEXT_NODE: Short = val
        val CDATA_SECTION_NODE: Short = val ENTITY_REFERENCE_NODE: Short = val
        val ENTITY_NODE: Short = val
        val PROCESSING_INSTRUCTION_NODE: Short = val
        val COMMENT_NODE: Short = val
        val DOCUMENT_NODE: Short = val
        val DOCUMENT_TYPE_NODE: Short = val
        val DOCUMENT_FRAGMENT_NODE: Short = val
        val NOTATION_NODE: Short = val
        val DOCUMENT_POSITION_DISCONNECTED: Short = val
        val DOCUMENT_POSITION_PRECEDING: Short = val
        val DOCUMENT_POSITION_FOLLOWING: Short = val
        val DOCUMENT_POSITION_CONTAINS: Short = val
        val DOCUMENT_POSITION_CONTAINED_BY: Short = val
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short = Exposes the JavaScript
[HTMLBaseElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBaseElement) to Kotlin

*public external abstract class HTMLBaseElement : HTMLElement {

    companion object {

        val ELEMENT_NODE: Short = val ATTRIBUTE_NODE: Short = val
        val TEXT_NODE: Short = val
        val CDATA_SECTION_NODE: Short = val ENTITY_REFERENCE_NODE: Short = val
        val ENTITY_NODE: Short = val
        val PROCESSING_INSTRUCTION_NODE: Short = val
        val COMMENT_NODE: Short = val
        val DOCUMENT_NODE: Short = val
        val DOCUMENT_TYPE_NODE: Short = val
        val DOCUMENT_FRAGMENT_NODE: Short = val
        val NOTATION_NODE: Short = val
        val DOCUMENT_POSITION_DISCONNECTED: Short = val
        val DOCUMENT_POSITION_PRECEDING: Short = val
        val DOCUMENT_POSITION_FOLLOWING: Short = val
        val DOCUMENT_POSITION_CONTAINS: Short = val
        val DOCUMENT_POSITION_CONTAINED_BY: Short = val
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short = Exposes the JavaScript
[HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin

*public external abstract class HTMLLinkElement : HTMLElement, LinkStyle {

    companion object {

        val ELEMENT_NODE: Short = val ATTRIBUTE_NODE: Short = val
        val TEXT_NODE: Short = val
        val CDATA_SECTION_NODE: Short = val ENTITY_REFERENCE_NODE: Short = val
        val ENTITY_NODE: Short = val
        val PROCESSING_INSTRUCTION_NODE: Short = val
        val COMMENT_NODE: Short = val
        val DOCUMENT_NODE: Short = val
        val DOCUMENT_TYPE_NODE: Short = val
        val DOCUMENT_FRAGMENT_NODE: Short = val
        val NOTATION_NODE: Short = val
        val DOCUMENT_POSITION_DISCONNECTED: Short = val
        val DOCUMENT_POSITION_PRECEDING: Short = val
        val DOCUMENT_POSITION_FOLLOWING: Short = val
        val DOCUMENT_POSITION_CONTAINS: Short = val
        val DOCUMENT_POSITION_CONTAINED_BY: Short = val
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short = Exposes the JavaScript

DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/** Exposes the JavaScript [HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin */
public external abstract class HTMLMetaElement : HTMLElement {
    open var name: String
    open var httpEquiv: String
    open var content: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/** Exposes the JavaScript [HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin */
public external abstract class HTMLStyleElement : HTMLElement, LinkStyle {
    open var media: String
    open var nonce: String
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/** Exposes the JavaScript [HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin */
public external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {
    open var text: String
    open var link: String
    open var vLink: String
    open var aLink: String
    open var bgColor: String
    open var background: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/** Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin */
public external abstract class HTMLHeadingElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

.rawValue

/**
 * Exposes the JavaScript
[HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin
 */
public external abstract class HTMLParagraphElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin
 */
public external abstract class HTMLHRElement : HTMLElement {
    open var align: String
    open var color: String
    open var noShade: Boolean
    open var size: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLPreElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPreElement) to Kotlin
 */
public external abstract class HTMLPreElement : HTMLElement {
    open var width: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLQuoteElement](https://developer.mozilla.org/en/docs/Web/API/HTMLQuoteElement) to Kotlin
 */
public external abstract class HTMLQuoteElement : HTMLElement {
    open var cite: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript
 [HTMLListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLListElement) to Kotlin
 */
public external abstract class HTMLOListElement : HTMLElement {
    open var reversed: Boolean
    open var start: Int
    open var type: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 [HTMLUListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLUListElement) to Kotlin
 */
public external abstract class HTMLUListElement : HTMLElement {
    open var compact: Boolean
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 [HTMLLIElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLIElement) to Kotlin
 */
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript
 [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin
 */
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
null
/**
 * Exposes the JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin */
public external abstract class HTMLBRElement : HTMLElement {
    open var clear: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to Kotlin
 */
public external interface HTMLHyperlinkElementUtils {
    var href: String
    val origin: String
    var protocol: String
    var username: String
    var password: String
    var host: String
    var hostname: String
    var port: String
    var pathname: String
    var search: String
    var hash: String
}

/**
 * Exposes the JavaScript [HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin
 */
public external abstract class HTMLModElement : HTMLElement {
    open var cite: String
    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLPictureElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement) to Kotlin
 */
public external abstract class HTMLPictureElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSourceElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement) to Kotlin
 */
```kotlin
/**
 * Exposes the JavaScript [HTMLImageElement](https://developer.mozilla.org/en/docs/Web/API/HTMLImageElement) to Kotlin
 */
public external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {
    open var alt: String
    open var src: String
    open var srcset: String
    open var sizes: String
    open var crossOrigin: String?
    open var useMap: String
    open var isMap: Boolean
    open var width: Int
    open var height: Int
    open val naturalWidth: Int
    open val naturalHeight: Int
    open val complete: Boolean
    open val currentSrc: String
    open var referrerPolicy: String
    open var name: String
    open var lowsrc: String
    open var align: String
    open var hspace: Int
    open var vspace: Int
    open var longDesc: String
    open var border: String
    open val x: Int
    open val y: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin
 */
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String
    open var srcdoc: String
    open var name: String
    open val sandbox: DOMTokenList
    open var allowFullscreen: Boolean
    open var allowUserMedia: Boolean
    open var width: String
    open var height: String
    open var referrerPolicy: String
    open var srcdoc: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var align: String
    open var scrolling: String
    open var frameborder: String
    open var longDesc: String
    open var marginHeight: String
    open var marginWidth: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin
 */
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String
    open var type: String
    open var width: String
    open var height: String
    open var align: String
    open var name: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```
Open Source Used In DNAC Platform Arica 13380

```kotlin
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin
 */
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String
    open var type: String
    open var typeMustMatch: Boolean
    open var name: String
    open var useMap: String
    open var form: HTMLFormElement?
    open var width: String
    open var height: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var align: String
    open var archive: String
    open var code: String
    open var declare: Boolean
    open var hspace: Int
    open var standby: String
    open var vspace: Int
    open var codeBase: String
    open var codeType: String
    open var border: String
    fun getSVGDocument(): Document?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin
 */
public external abstract class HTMLParamElement : HTMLElement {
    open var name: String
    open var value: String
    open var valueType: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin
 */
public external abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    open var videoWidth: Int
    open var videoHeight: Int
    open var poster: String
    open var playsInline: Boolean

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val HAVE_NOTHING: Short
        val HAVE_META DATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
```
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin
 */
public external abstract class HTMLAudioElement : HTMLMediaElement {
    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin
 */
public external abstract class HTMLTrackElement : HTMLElement {
    open var kind: String
    open var src: String
    open var srclang: String
    open var label: String
    open var default: Boolean
    open val readyState: Short
    open val track: TextTrack

    companion object {
        val NONE: Short
        val LOADING: Short
        val LOADED: Short
        val ERROR: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to Kotlin
 */
public external abstract class HTMLMediaElement : HTMLElement {
    open val error: MediaError?
    open var src: String
    open var srcObject: dynamic
    open val currentSrc: String
    open var crossOrigin: String?
    open val networkState: Short
    open var preload: String
    open val buffered: TimeRanges
    open val readyState: Short
    open var seeking: Boolean
    open var currentTime: Double
    open val duration: Double
    open var paused: Boolean
    open var defaultPlaybackRate: Double
    open var playbackRate: Double
    open val played: TimeRanges
    open val seekable: TimeRanges
    open var ended: Boolean
    open var autoplay: Boolean
    open var loop: Boolean
    open var controls: Boolean
    open var volume: Double
    open var muted: Boolean
    open var defaultMuted: Boolean
    open val audioTracks: AudioTrackList
    open val videoTracks: VideoTrackList
    open val textTracks: TextTrackList
    fun load()
    fun canPlayType(type: String): CanPlayTypeResult
    fun fastSeek(time: Double)
    fun getStartDate(): dynamic
    fun play(): Promise<Unit>
    fun pause()
    fun addTextTrack(kind: TextTrackKind, label: String = definedExternally,
language: String = definedExternally): TextTrack

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE NOTHING: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
 [MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin
 */
public external abstract class MediaError {
    open val code: Short

    companion object {
        val MEDIA_ERR_ABORTED: Short
        val MEDIA_ERR_NETWORK: Short
        val MEDIA_ERR_DECODE: Short
        val MEDIA_ERR_SRC_NOT_SUPPORTED: Short
    }
}

/**
 * Exposes the JavaScript
 [AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin
 */
public external abstract class AudioTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): AudioTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun AudioTrackList.get(index: Int): AudioTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin
 */
public external abstract class VideoTrackList : EventTarget {
    open val length: Int
    open val selectedIndex: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): VideoTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun VideoTrackList.get(index: Int): VideoTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [TextTrackList](https://developer.mozilla.org/en/docs/Web/API/TextTrackList) to Kotlin
 */
public external abstract class TextTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): TextTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackList.get(index: Int): TextTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript
 [TextTrackCueList](https://developer.mozilla.org/en/docs/Web/API/TextTrackCueList) to Kotlin
 */
public external abstract class TextTrackCueList {
    open val length: Int
    fun getCueById(id: String): TextTrackCue?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackCueList.get(index: Int): TextTrackCue? = asDynamic()[index]
abstract class TextTrackCue : EventTarget {
    open val open var track: TextTrack?
    open var open var startTime: Double
    open var open var endTime: Double
    open var open var pauseOnExit: Boolean
    open val open var onenter: ((Event) -> dynamic)?
    open val open var onexit: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript
[TimeRanges](https://developer.mozilla.org/en/docs/Web/API/TimeRanges) to Kotlin
 */
public external abstract class TimeRanges {
    open val length: Int
    fun start(index: Int): Double
    fun end(index: Int): Double
}

/**
 * Exposes the JavaScript
[TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin
 */
public external open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {
    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface TrackEventInit : EventInit {
    var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["track"] = track
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin
 */
public external abstract class HTMLMapElement : HTMLElement {
    open var name: String
    open val areas: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin
 */
public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var alt: String
    open var coords: String
    open var shape: String
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var referrerPolicy: String
    open var noHref: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var caption: HTMLTableCaptionElement?
    open var tHead: HTMLTableSectionElement?
    open var tFoot: HTMLTableSectionElement?
    open var tBodies: HTMLCollection
    open var tRows: HTMLCollection
    open var tAlign: String
    open var tOpenVarBorder: String
    open var tOpenVarFrame: String
    open var tOpenVarRules: String
    open var tOpenVarSummary: String
    open var tOpenVarWidth: String
    open var bgColor: String
    open var cellPadding: String
    open var
var cellSpacing: String
fun createCaption(): HTMLTableCaptionElement
fun deleteCaption(): HTMLTableCaptionElement
fun createThead(): HTMLTableSectionElement
fun deleteThead(): HTMLTableSectionElement
fun createTfoot(): HTMLTableSectionElement
fun deleteTfoot(): HTMLTableSectionElement

fun insertRow(index: Int = definedExternally): HTMLTableRowElement
fun deleteRow(index: Int)

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin
 */
public external abstract class HTMLTableCaptionElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin
 */
public external abstract class HTMLTableColElement : HTMLElement {
    open var span: Int
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin
 */
public external abstract class HTMLTableSectionElement : HTMLElement {
    open val rows: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    fun insertRow(index: Int = definedExternally): HTMLElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
 /**
 * Exposes the JavaScript [HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin
 */
open external abstract class HTMLTableRowElement : HTMLElement {
    open val rowIndex: Int
    open val sectionRowIndex: Int
    open val cells: HTMLCollection
    open var align: String
    open var ch: String
    open var chOff: String
    open var vAlign: String
    open var bgColor: String
    fun insertCell(index: Int = definedExternally): HTMLElement
    fun deleteCell(index: Int)
}

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
}

/**
 * Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin
 */
open external abstract class HTMLTableCellElement : HTMLElement {
    open var colSpan: Int
    open var rowSpan: Int
    open var headers: String
    open val cellIndex: Int
    open var scope: String
    open var abbr: String
    open var align: String
    open var axis: String
    open var height: String
    open var width: String
    open var ch: String
    open var chOff: String
    open var noWrap: Boolean
    open var vAlign: String
    open var bgColor: String
}

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
}

/**
 * Exposes the JavaScript [HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin
 */
public external abstract class HTMLFormElement : HTMLElement {
    open var acceptCharset: String
    open var action: String
    open var autocomplete: String
    open var enctype: String
    open var encoding: String
    open var method: String
    open var name: String
    open var noValidate: Boolean
    open var target: String
    open val elements: HTMLFormControlsCollection
    open val length: Int
    fun submit()
    fun reset()
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(name: String): UnionElementOrRadioNodeList? = asDynamic()[name]
Open Source Used In DNAC Platform Arica

[HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin

```kotlin
external abstract class HTMLLabelElement : HTMLElement {
    open val form: HTMLFormElement?
    open val htmlFor: String
    open val control: HTMLElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

[HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin

```kotlin
external abstract class HTMLInputElement : HTMLElement {
    open var accept: String
    open var alt: String
    open var autocomplete: String
    open var autofocus: Boolean
    open var defaultChecked: Boolean
    open var checked: Boolean
    open var dirName: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open val files: FileList?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var height: Int
    open var indeterminate: Boolean
    open var inputMode: String
    open val list: HTMLElement?
    open var max: String
    open var maxLength: Int
    open var min: String
    open var minLength: Int
    open var multiple: Boolean
    open var name: String
    open var pattern: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var size: Int
    open var src: String
    open var step: String
    open var type: String
    open var defaultValue: String
    open var value: String
    open var valueAsDate: dynamic
    open var valueAsNumber: Double
    open var width: Int
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var labels: NodeList
    open var selectionStart: Int?
    open var selectionEnd: Int?
    open var selectionDirection: String
    fun stepUp(n: Int = definedExternally)
    fun stepDown(n: Int = definedExternally)
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    fun select()
    fun setRangeText(replacement: String)
    fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
    fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

[HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin

```kotlin
external abstract class HTMLButtonElement : HTMLElement {
    open var autofocus: Boolean
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var menu: HTMLMenuElement?
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var labels: NodeList
    open var selectionStart: Int
    open var selectionEnd: Int
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
```
fun setCustomValidity(error: String)

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin
 */
public external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {
    open var autocomplete: String
    open var autofocus: Boolean
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var multiple: Boolean
    open var name: String
    open var required: Boolean
    open var size: Int
    open val type: String
    open val options: HTMLOptionsCollection
    override var length: Int
    open val selectedOptions: HTMLCollection
    open var selectedIndex: Int
    open var value: String
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    fun namedItem(name: String): HTMLOptionElement?
    fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
    fun remove(index: Int)
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    override fun item(index: Int): Element?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLSelectElement.set(index: Int, option: HTMLOptionElement?) = asDynamic()[index] = option

/**
 * Exposes the JavaScript [HTMLDataListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataListElement) to Kotlin
 */
public external abstract class HTMLDataListElement : HTMLElement {
    open val options: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLOptGroupElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptGroupElement) to Kotlin
 */
public external abstract class HTMLOptGroupElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
    open var disabled: Boolean
    open var label: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
/**
 * Exposes the JavaScript [HTMLOptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOptionElement) to Kotlin
 */
public external abstract class HTMLOptionElement : HTMLElement, UnionHTMLOptGroupElementOrHTMLOptionElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var label: String
    open var defaultSelected: Boolean
    open var selected: Boolean
    open var value: String
    open var text: String
    open val index: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTextAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTextAreaElement) to Kotlin
 */
public external abstract class HTMLTextAreaElement : HTMLElement {
    open var autocomplete: String
    open var autofocus: Boolean
    open var cols: Int
    open var dirName: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var inputMode: String
    open var maxLength: Int
    open var minLength: Int
    open var name: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var rows: Int
    open var wrap: String
    open val type: String
    open var defaultValue: String
    open var value: String
    open val textLength: Int
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    open val labels: NodeList
    open var selectionStart: Int?
    open var selectionEnd: Int?
    open var selectionDirection: String?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)
    fun select()
    fun setRangeText(replacement: String)
    fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
    fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLKeygenElement](https://developer.mozilla.org/en/docs/Web/API/HTMLKeygenElement) to Kotlin
 */
public external abstract class HTMLKeygenElement : HTMLElement {
    open var autofocus: Boolean
    open var challenge: String
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var
keytype: String\n  open var name: String\n  open val type: String\n  open val willValidate: Boolean\n  open val validity: ValidityState\n  open val validationMessage: String\n  open val labels: NodeList\n  fun checkValidity(): Boolean\n  fun reportValidity(): Boolean\n  fun setCustomValidity(error: String)\n
  companion object {
    val ELEMENT_NODE: Short\n    val ATTRIBUTE_NODE: Short\n    val TEXT_NODE: Short\n    val CDATA_SECTION_NODE: Short\n    val ENTITY_REFERENCE_NODE: Short\n    val ENTITY_NODE: Short\n    val PROCESSING_INSTRUCTION_NODE: Short\n    val COMMENT_NODE: Short\n    val DOCUMENT_NODE: Short\n    val DOCUMENT_TYPE_NODE: Short\n    val DOCUMENT_FRAGMENT_NODE: Short\n    val DOCUMENT_POSITION_DISCONNECTED: Short\n    val DOCUMENT_POSITION_PRECEDING: Short\n    val DOCUMENT_POSITION_FOLLOWING: Short\n    val DOCUMENT_POSITION_CONTAINS: Short\n    val DOCUMENT_POSITION_CONTAINED_BY: Short\n    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n  }

  /**
   * Exposes the JavaScript [HTMLOutputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLOutputElement) to Kotlin
   */
  public external abstract class HTMLOutputElement : HTMLElement {
    open var htmlFor: DOMTokenList\n    open var form: HTMLFormElement?\n    open var name: String\n    open var defaultValue: String\n    open var value: String\n    open val willValidate: Boolean\n    open val validity: ValidityState\n    open val validationMessage: String\n    open val labels: NodeList\n    fun checkValidity(): Boolean\n    fun reportValidity(): Boolean\n    fun setCustomValidity(error: String)

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val TEXT_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_NODE: Short
      val DOCUMENT_TYPE_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
      val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
  }

  /**
   * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin
   */
  public external abstract class HTMLProgressElement : HTMLElement {
    open var value: Double\n    open var max: Double\n    open var position: Double\n    open val labels: NodeList

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val TEXT_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_NODE: Short
      val DOCUMENT_TYPE_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
  }

  /**
   * Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin
   */
  public external abstract class HTMLMeterElement : HTMLElement {
    open var value: Double\n    open var min: Double\n    open var max: Double\n    open var optimum: Double

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val TEXT_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_NODE: Short
      val DOCUMENT_TYPE_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

Open Source Used In DNAC Platform Arica  13389
ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLTemplateElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTemplateElement) to Kotlin
 */
public external abstract class HTMLTemplateElement : HTMLElement {
    open val content: DocumentFragment

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSlotElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSlotElement) to Kotlin
 */
public external abstract class HTMLSlotElement : HTMLElement {
    open var name: String
    fun assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface AssignedNodesOptions {
    var flatten: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun AssignedNodesOptions(flatten: Boolean? = false): AssignedNodesOptions {
    val o = js("({})")
    o["flatten"] = flatten
    return o
}

/**
 * Exposes the JavaScript [HTMLCanvasElement](https://developer.mozilla.org/en/docs/Web/API/HTMLCanvasElement) to Kotlin
 */
public external abstract class HTMLCanvasElement : HTMLElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    fun getContext(contextId: String, vararg arguments: Any?): RenderingContext?
    fun toDataURL(type: String = definedExternally, quality: Any? = definedExternally): String
    fun toBlob(_callback: (Blob?) -> Unit, type: String = definedExternally, quality: Any? = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val alpha: Boolean? /* = true */
set(value) = definedExternally

fun createLinearGradient(x0: Double, y0: Double, x1: Double, y1: Double): CanvasGradient
fun createRadialGradient(x0: Double, y0: Double, r0: Double, x1: Double, y1: Double, r1: Double): CanvasGradient
fun createPattern(image: CanvasImageSource, repetition: String?): CanvasPattern?
fun createRadialGradient(x0: Double, y0: Double, r0: Double, x1: Double, y1: Double, r1: Double): CanvasGradient
fun createPattern(image: CanvasImageSource, repetition: String?): CanvasPattern?

fun clearRect(x: Double, y: Double, w: Double, h: Double)
fun fillRect(x: Double, y: Double, w: Double, h: Double)
fun strokeRect(x: Double, y: Double, w: Double, h: Double)

fun beginPath()
fun fill(fillRule: CanvasFillRule = definedExternally)
fun stroke()
fun clip(fillRule: CanvasFillRule = definedExternally)
fun fillText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)
fun strokeText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)
fun paintRect(x: Double, y: Double, w: Double, h: Double)
fun paintPath(path: Path2D)
fun paintStrokePath(path: Path2D)

window?

Open Source Used In DNAC Platform Arica 13396
Open Source Used In DNAC Platform Arica  13397

[PopStateEvent](https://developer.mozilla.org/en/docs/Web/API/PopStateEvent) to Kotlin

```kotlin
open external class PopStateEvent(type: String, eventInitDict: PopStateEventInit = definedExternally) : Event {
    open var state: Any?
}
```

```kotlin
companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
}
```

```kotlin
public external interface PopStateEventInit : EventInit {
    var state: Any? /* = null */
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["state"] = state
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["return"] = return
    return o
}
```

[HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin

```kotlin
open external class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) : Event {
    open val oldURL: String
    open val newURL: String
}
```

```kotlin
companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
}
```

```kotlin
public external interface HashChangeEventInit : EventInit {
    var oldURL: String? /* = "" */
    var newURL: String? /* = "" */
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["oldURL"] = oldURL
    o["newURL"] = newURL
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["return"] = return
    return o
}
```

[PageTransitionEvent](https://developer.mozilla.org/en/docs/Web/API/PageTransitionEvent) to Kotlin

```kotlin
open external class PageTransitionEvent(type: String, eventInitDict: PageTransitionEventInit = definedExternally) : Event {
    open val persisted: Boolean
}
```

```kotlin
companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
}
```

```kotlin
public external interface PageTransitionEventInit : EventInit {
    var persisted: Boolean? /* = false */
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["persisted"] = persisted
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    o["return"] = return
    return o
}
```

[BeforeUnloadEvent](https://developer.mozilla.org/en/docs/Web/API/BeforeUnloadEvent) to Kotlin

```kotlin
class ApplicationCache : EventTarget {
    open var status: Short
    open var onchecking: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onnnoupdate: ((Event) -> dynamic)?
    open var ondownloading: ((Event) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onupdateready: ((Event) -> dynamic)?
    open var oncached: ((Event) -> dynamic)?
    open var onobsolete: ((Event) -> dynamic)?
}
```

```kotlin
class NavigatorOnLine {
    val onLine: Boolean
}
```

```kotlin
class ErrorEvent(type: String, eventInitDict: ErrorEventInit = definedExternally) : Event {
    open var message: String
    open var filename: String
    open var lineno: Int
    open var colno: Int
    open var error: Any?
}
```

```kotlin
companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
}
```

```kotlin
public external interface ErrorEventInit : EventInit {
    var message: String
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun ErrorEventInit(message: String? = null): ErrorEventInit {
    val o = js("({})")
    o["message"] = message
    return o
}
```
String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var filename: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var lineno: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var colno: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var error: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["message"] = message
    o["filename"] = filename
    o["lineno"] = lineno
    o["colno"] = colno
    o["error"] = error
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[PromiseRejectionEvent](https://developer.mozilla.org/en/docs/Web/API/PromiseRejectionEvent) to Kotlin
 */
public external open class PromiseRejectionEvent(type: String, eventInitDict: PromiseRejectionEventInit) : Event {
    open val promise: Promise<Any?>
    open val reason: Any?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface PromiseRejectionEventInit : EventInit {
        var promise: Promise<Any?>?
            get() = definedExternally
            set(value) = definedExternally
        var reason: Any?
            get() = definedExternally
            set(value) = definedExternally
    }

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["promise"] = promise
    o["reason"] = reason
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
[GlobalEventHandlers](https://developer.mozilla.org/en/docs/Web/API/GlobalEventHandlers) to Kotlin
 */
public external interface GlobalEventHandlers {
    var onabort: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onblur: ((FocusEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncancel: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncanplay: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncanplaythrough: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onchange: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onclick: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onclose: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally

definedExternally
        set(value) = definedExternally
    var onpointerleave: ((PointerEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
[WindowEventHandlers](https://developer.mozilla.org/en/docs/Web/API/WindowEventHandlers) to Kotlin
*/
public external interface WindowEventHandlers {
    var onafterprint: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onbeforeprint: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onbeforeunload: ((BeforeUnloadEvent) -> String)?
        get() = definedExternally
        set(value) = definedExternally
    var onhashchange: ((HashChangeEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onlanguagechange: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onmessage: ((MessageEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onoffline: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var ononline: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpagehide: ((PageTransitionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpageshow: ((PageTransitionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpopstate: ((PopStateEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onrejectionhandled: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onstorage: ((StorageEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onunload: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

public external interface DocumentAndElementEventHandlers {
    var oncopy: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var oncut: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
    var onpaste: ((ClipboardEvent) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript
[WindowOrWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WindowOrWorkerGlobalScope) to Kotlin
*/
public external interface WindowOrWorkerGlobalScope {
    val origin: String
    val caches: CacheStorage
    fun btoa(data: String): String
    fun atob(data: String): String
    fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearTimeout(handle: Int = definedExternally)
    fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearInterval(handle: Int = definedExternally)
    fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>
    fun fetch(input: dynamic, init: RequestInit = definedExternally): Response
}

/**
 * Exposes the JavaScript
[Navigator](https://developer.mozilla.org/en/docs/Web/API/Navigator) to Kotlin
*/
public external abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, Navigator.ConcurrentHardware {
    open val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val maxTouchPoints: Int
    open val serviceWorker: ServiceWorkerContainer
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

/**
 * Exposes the JavaScript
[NavigatorID](https://developer.mozilla.org/en/docs/Web/API/NavigatorID) to Kotlin
*/
public external interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val oscpu: String
    fun taintEnabled(): Boolean
}

### Open Source Used In DNAC Platform Arica 13400
fun registerContentHandler(mimeType: String, url: String, title: String) {
    isProtocolHandlerRegistered(scheme: String, url: String): String
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

/**
 * Exposes the JavaScript [NavigatorPlugins](https://developer.mozilla.org/en/docs/Web/API/NavigatorPlugins) to Kotlin *
 */
public external interface NavigatorPlugins {
    val pluginArray: PluginArray
    fun refresh(reload: Boolean = definedExternally)
}

/**
 * Exposes the JavaScript [PluginArray](https://developer.mozilla.org/en/docs/Web/API/PluginArray) to Kotlin *
 */
public external abstract class PluginArray : ItemArrayLike<Plugin> {
    override fun item(index: Int): Plugin?
    fun namedItem(name: String): Plugin?
}

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(index: Int): Plugin? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]

/**
 * Exposes the JavaScript [MimeTypeArray](https://developer.mozilla.org/en/docs/Web/API/MimeTypeArray) to Kotlin *
 */
public external abstract class MimeTypeArray : ItemArrayLike<MimeType> {
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
}

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(index: Int): MimeType? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(name: String): MimeType? = asDynamic()[name]

/**
 * Exposes the JavaScript [MimeType](https://developer.mozilla.org/en/docs/Web/API/MimeType) to Kotlin *
 */
public external abstract class MimeType {
    open val type: String
    open val description: String
    open val suffixes: String
    open val enabledPlugin: Plugin
}

/**
 * Exposes the JavaScript [ImageBitmap](https://developer.mozilla.org/en/docs/Web/API/ImageBitmap) to Kotlin *
 */
public external abstract class ImageBitmap : CanvasImageSource, TexImageSource {
    open val width: Int
    open val height: Int
    fun close()
}

public external interface ImageBitmapOptions {
    var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */
    var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */
    var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */
    var resizeWidth: Int?
    var resizeHeight: Int?
    var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */
    var set(value: Int): ImageBitmapOptions
    var get(): ImageBitmapOptions
}

@kotlin.internal.InlineOnly
public inline fun ImageBitmapOptions(imageOrientation: ImageOrientation? = ImageOrientation.NONE,
premultiplyAlpha: PremultiplyAlpha? = PremultiplyAlpha.DEFAULT,
colorSpaceConversion: ColorSpaceConversion? = ColorSpaceConversion.DEFAULT,
resizeWidth: Int? = undefined,
resizeHeight: Int? = undefined,
resizeQuality: ResizeQuality? = ResizeQuality.LOW): ImageBitmapOptions {
    val o = js("({})")
    o["imageOrientation"] = imageOrientation
    o["premultiplyAlpha"] = premultiplyAlpha
    o["colorSpaceConversion"] = colorSpaceConversion
    o["resizeWidth"] = resizeWidth
    o["resizeHeight"] = resizeHeight
    o["resizeQuality"] = resizeQuality
    return o
}

/**
 * Exposes the JavaScript [MessageEvent](https://developer.mozilla.org/en/docs/Web/API/MessageEvent) to Kotlin *
 */
public external open class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {
    open val data: String
}
Any?

open val origin: String?
open val lastEventId: String?
open val source: UnionMessagePortOrWindowProxy?
open val ports: Array<out MessagePort>

fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)

companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
    val UNCAPTURING_PHASE: Short
}

public external interface MessageEventInit : EventInit {
    var data: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var origin: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var lastEventId: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var source: UnionMessagePortOrWindowProxy? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [EventSource](https://developer.mozilla.org/en/docs/Web/API/EventSource) to Kotlin
 */
public external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
    open val url: String
    open val withCredentials: Boolean
    open val readyState: Short
    var onopen: ((Event) -> dynamic)?
    var onmessage: ((MessageEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    fun close()

    companion object {
        val CONNECTING: Short
        val OPEN: Short
        val CLOSED: Short
    }
}

public external interface EventSourceInit {
    var withCredentials: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
    val o = js("({})")
    o["withCredentials"] = withCredentials
    return o
}

/**
 * Exposes the JavaScript [WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin
 */
public external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
    open val url: String
    open val readyState: Short
    open val bufferedAmount: Number
    var onopen: ((Event) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    var onclose: ((Event) -> dynamic)?
    open val extensions: String
    open val protocol: String
    var onmessage: ((MessageEvent) -> dynamic)?
    var binaryType: BinaryType
    fun close(code: Short = definedExternally, reason: String = definedExternally)
    fun send(data: String)
    fun send(data: Blob)
    fun send(data: ArrayBuffer)
    fun send(data: ArrayBufferView)

    companion object {
        val CONNECTING: Short
        val OPEN: Short
        val CLOSING: Short
        val CLOSED: Short
    }
}

/**
 * Exposes the JavaScript [CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin
 */
public external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {
    open val wasClean: Boolean
    open val code: Short
    open val reason: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CloseEventInit : EventInit {
    var wasClean: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var code: Short? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var reason: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["wasClean"] = wasClean
    o["code"] = code
    o["reason"] = reason
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin
 */
public external open class MessageChannel {
    open val port1: MessagePort
    open val port2: MessagePort
}
Exposes the JavaScript [MessagePort](https://developer.mozilla.org/en/docs/Web/API/MessagePort) to Kotlin

```kotlin
*//public external abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy,
UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker { 
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()
}

/**
 * Exposes the JavaScript [BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin
 */
public external open class BroadcastChannel(name: String) : EventTarget {
    var name: String
    fun postMessage(message: Any?)
    fun close()
}

/**
 * Exposes the JavaScript [WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin
 */
public external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope,
GlobalPerformance {
    val self: WorkerGlobalScope
    val location: WorkerLocation
    val navigator: WorkerNavigator
    var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    var onlanguagechange: ((Event) -> dynamic)?
    var onoffline: ((Event) -> dynamic)?
    var ononline: ((Event) -> dynamic)?
    var onrejectionhandled: ((Event) -> dynamic)?
    var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
    fun importScripts(vararg urls: String)
}

/**
 * Exposes the JavaScript [AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin
 */
public external interface AbstractWorker {
    var onerror: ((Event) -> dynamic)?
        get() = definedExternally
        set(value) = definedExternally
}

/**
 * Exposes the JavaScript [Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin
 */
public external open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    var onmessage: ((MessageEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    fun terminate()
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

public external interface WorkerOptions {
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
    var credentials: RequestCredentials? /* = RequestCredentials.OMIT */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun WorkerOptions(type: WorkerType? = WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {
    val o = js("({})")
    o["type"] = type
    o["credentials"] = credentials
    return o
}

/**
 * Exposes the JavaScript [SharedWorker](https://developer.mozilla.org/en/docs/Web/API/SharedWorker) to Kotlin
 */
public external open class SharedWorker(scriptURL: String, name: String = definedExternally, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    open val port: MessagePort
    override var onerror: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript [NavigatorConcurrentHardware](https://developer.mozilla.org/en/docs/Web/API/NavigatorConcurrentHardware) to Kotlin
 */
public external interface NavigatorConcurrentHardware {
    val hardwareConcurrency: Number
}

/**
 * Exposes the JavaScript [WorkerNavigator](https://developer.mozilla.org/en/docs/Web/API/WorkerNavigator) to Kotlin
 */
public external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine,
NavigatorConcurrentHardware {
    val serviceWorker: ServiceWorkerContainer
}
```
```kotlin
/*
 * Exposes the JavaScript [Storage](https://developer.mozilla.org/en/docs/Web/API/Storage) to Kotlin's *
 * abstract external class Storage:
 * type: String
 * eventInitDict: StorageEventInit = definedExternally) : Event {
 * open var key: String?
 * open val oldValue: String?
 * open val newValue: String?
 * open val url: String?
 * open val bubbles: Boolean?
 * open val cancelable: Boolean?
 * 
 * @kotlin.internal.InlineOnly
 * public inline fun StorageEventInit(key: String? = null,
 * 
 * public external abstract class WorkerLocation {
 * 
 * }*
 */

```
CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin
 */
public external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {
    open var cols: String
    open var rows: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLFrameElement : HTMLElement {
    open var name: String
    open var scrolling: String
    open var src: String
    open var frameBorder: String
    open var longDesc: String
    open var noResize: Boolean
    open val contentDocument: Document?
    open val contentWindow: Window?
    open var marginHeight: String
    open var marginWidth: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLDirectoryElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [HTMLFontElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFontElement) to Kotlin
 */
public external abstract class HTMLFontElement : HTMLElement {
    open var color: String
    open var face: String
    open var size: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
The document contains Java code related to event handling and document manipulation, including interfaces for event init, custom event, and non-element parent node. It also references JavaScript methods from Mozilla developer documentation, showing how these methods can be used in Kotlin for JavaScript interop. The text includes method signatures, class definitions, and code snippets that demonstrate the use of these features in Kotlin.
Open Source Used In DNAC Platform Arica 13407

[Slatable](https://developer.mozilla.org/en/docs/Web/API/Slatable) to Kotlin

```kotlin
public external interface Slotable {
    val assignedSlot: HTMLSlotElement?
        get() = definedExternally
        set(value) = definedExternally
}
```

[Slotable](https://developer.mozilla.org/en/docs/Web/API/Slatable) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
 *//*public external abstract class NodeList : ItemArrayLike<Node> {
    fun namedItem(name: String): Node?
}

@kotlin.internal.InlineOnly
```

[NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [HTMLCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLCollection) to Kotlin
 */
public external abstract class HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {
    fun namedItem(name: String): Element?
}
```

[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtree: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}
```

[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin
 */
public external abstract class MutationRecord {
    var type: String
    var target: Node
    var addedNodes: NodeList
    var removedNodes: NodeList
    var previousSibling: Node?
    var nextSibling: Node?
    var attributeName: String?
    var attributeNamespace: String?
    var oldValue: String?
}
```

[Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
 */
public external abstract class NodeList : ItemArrayLike<Node> {
    fun namedItem(name: String): Node?
}
```

```kotlin
/**
 * Exposes the JavaScript
 * [HTMLCollection](https://developer.mozilla.org/en/docs/Web/API/HTMLCollection) to Kotlin
 *//*public external abstract class HTMLCollection : ItemArrayLike<Element>, UnionElementOrHTMLCollection {
    fun namedItem(name: String): Element?
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(index: Int): Element? = asDynamic()[index]
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(name: String): Element? = asDynamic()[name]
```

[MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin
 */
public external abstract class MutationRecord {
    var type: String
    var target: Node
    var addedNodes: NodeList
    var removedNodes: NodeList
    var previousSibling: Node?
    var nextSibling: Node?
    var attributeName: String?
    var attributeNamespace: String?
    var oldValue: String?
}
```

[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtree: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    o["subtree"] = subtree
    o["attributeOldValue"] = attributeOldValue
    o["characterDataOldValue"] = characterDataOldValue
    o["attributeFilter"] = attributeFilter
    return o
}
```

[MutationObserver](https://developer.mozilla.org/en/docs/Web/API/MutationObserver) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtree: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    o["subtree"] = subtree
    o["attributeOldValue"] = attributeOldValue
    o["characterDataOldValue"] = characterDataOldValue
    o["attributeFilter"] = attributeFilter
    return o
}
```

[MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin

```kotlin
/**
 * Exposes the JavaScript
 * [MutationObserverInit](https://developer.mozilla.org/en/docs/Web/API/MutationObserverInit) to Kotlin
 */
public external interface MutationObserverInit {
    var childList: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributes: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterData: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var subtree: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var characterDataOldValue: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var attributeFilter: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["childList"] = childList
    o["attributes"] = attributes
    o["characterData"] = characterData
    o["subtree"] = subtree
    o["attributeOldValue"] = attributeOldValue
    o["characterDataOldValue"] = characterDataOldValue
    o["attributeFilter"] = attributeFilter
    return o
}
```
ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external interface GetRootNodeOptions {
    var composed: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean? = false): GetRootNodeOptions {
    val o = js("({})")
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to Kotlin
 *
 */
public external open class Document : Node, GlobalEventHandlers, DocumentAndElementEventHandlers,
NonElementParentNode, DocumentOrShadowRoot, ParentNode, GeometryUtils {
    open val implementation: DOMImplementation
    open val URL: String
    open val documentURI: String
    open val origin: String
    open val compatMode: String
    open val characterSet: String
    open val charset: String
    open val inputEncoding: String
    open val contentType: String
    open val doctype: DocumentType?
    open val documentElement: Element?
    open val location: Location?
    var domain: String
    open val referrer: String
    var cookie: String
    open val lastModified: String
    open val readyState: DocumentReadyState
    var title: String
    var dir: String
    var body: HTMLElement?
    open val head: HTMLHeadElement?
    open val images: HTMLCollection
    open val embeds: HTMLCollection
    open val plugins: HTMLCollection
    open val links: HTMLCollection
    open val forms: HTMLCollection
    open val scripts: HTMLCollection
    open val currentScript: HTMLOrSVGScriptElement?
    open val defaultView: Window?
    open val activeElement: Element?
    var designMode: String
    var onreadystatechange: ((Event) -> dynamic)?
    var fgColor: String
    var linkColor: String
    var vlinkColor: String
    var alinkColor: String
    var bgColor: String
    open val anchors: HTMLCollection
    open val applets: HTMLCollection
    open val all: HTMLAllCollection
    open val scrollingElement: Element?
    open val styleSheets: StyleSheetList
    open val rootElement: SVGSVGElement
    open val fullscreenEnabled: Boolean
    open val fullscreen: Boolean
    var onfullscreenchange: ((Event) -> dynamic)?
    var onfullscreenerror: ((Event) -> dynamic)?
    override var onabort: ((Event) -> dynamic)?
    override var onblur: ((FocusEvent) -> dynamic)?
    override var oncancel: ((Event) -> dynamic)?
    override var oncanplay: ((Event) -> dynamic)?
    override var oncanplaythrough: ((Event) -> dynamic)?
    override var onclose: ((Event) -> dynamic)?
    override var oncontextmenu: ((MouseEvent) -> dynamic)?
    override var oncuechange: ((Event) -> dynamic)?
    override var ondblclick: ((MouseEvent) -> dynamic)?
    override var ondrag: ((DragEvent) -> dynamic)?
    override var ondragend: ((DragEvent) -> dynamic)?
    override var ondragenter: ((DragEvent) -> dynamic)?
    override var ondragleave: ((DragEvent) -> dynamic)?
    override var ondragover: ((DragEvent) -> dynamic)?
    override var ondragstart: ((DragEvent) -> dynamic)?
    override var ondurationchange: ((Event) -> dynamic)?
    override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    override var onfocus: ((FocusEvent) -> dynamic)?
    override var onkeydown: ((KeyboardEvent) -> dynamic)?
    override var onkeypress: ((KeyboardEvent) -> dynamic)?
    override var onkeyup: ((KeyboardEvent) -> dynamic)?
    override var onmessage: ((message: MessageEvent) -> dynamic)?
    override var onunhandled: ((message: MessageEvent) -> dynamic)?
    override var onsubmit: ((Event) -> dynamic)?
    override var onunload: ((Event) -> dynamic)?
}
`createTouchList(vararg touches: Touch): TouchList

``
fun exitFullscreen(): Promise<Unit>
``

`override fun
getElementById(elementId: String): Element?`

`override fun append(vararg nodes: dynamic)
``

`override fun querySelector(selectors: String): Element?`

`override fun querySelectorAll(selectors: String): NodeList`

`override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>`

`override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad`

`override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad`

`override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint`
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

EXPOSES THE JAVASCRIPT [DOCUMENT_FRAGMENT](https://developer.mozilla.org/en/docs/Web/API/DocumentFragment) TO KOTLIN


(val override fun prepend(vararg nodes: dynamic)

.override fun append(vararg nodes: dynamic)

.override fun querySelector(selectors: String): Element?

.override fun querySelectorAll(selectors: String)

.companion object {

.val ELEMENT_NODE: Short
.val ATTRIBUTE_NODE: Short
.val TEXT_NODE: Short
.val CDATA_SECTION_NODE: Short
.val ENTITY_REFERENCE_NODE: Short
.val ENTITY_NODE: Short
.val PROCESSING_INSTRUCTION_NODE: Short

.val DOCUMENT_FRAGMENT_NODE: Short
.val NOTATION_NODE: Short
.val DOCUMENT_POSITION_DISCONNECTED: Short
.val DOCUMENT_POSITION_PRECEDING: Short

.val DOCUMENT_POSITION_FOLLOWING: Short
.val DOCUMENT_POSITION_CONTAINS: Short
.val DOCUMENT_POSITION_CONTAINED_BY: Short
.val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

EXPOSES THE JAVASCRIPT [SHADOWROOT](https://developer.mozilla.org/en/docs/Web/API/ShadowRoot) TO KOTLIN

(val open external open class ShadowRoot : DocumentFragment, DocumentOrShadowRoot {

.open val mode: ShadowRootMode

.open val host: Element

.override val fullscreenElement: Element?

.companion object {

.val ELEMENT_NODE: Short
.val ATTRIBUTE_NODE: Short
.val TEXT_NODE: Short
.val CDATA_SECTION_NODE: Short
.val ENTITY_REFERENCE_NODE: Short
.val ENTITY_NODE: Short
.val PROCESSING_INSTRUCTION_NODE: Short
.val DOCUMENT_NODE: Short
.val DOCUMENT_TYPE_NODE: Short
.val DOCUMENT_FRAGMENT_NODE: Short
.val NOTATION_NODE: Short
.val DOCUMENT_POSITION_DISCONNECTED: Short
.val DOCUMENT_POSITION_PRECEDING: Short

.val DOCUMENT_POSITION_FOLLOWING: Short
.val DOCUMENT_POSITION_CONTAINS: Short
.val DOCUMENT_POSITION_CONTAINED_BY: Short
.val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

EXPOSES THE JAVASCRIPT [ELEMENT](https://developer.mozilla.org/en/docs/Web/API/Element) TO KOTLIN

(val open external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils,
UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent,
UnionElementOrProcessingInstruction {

.open val namespaceURI: String?
.open val prefix: String?
.open val localName: String
.open val tagName: String
.open var id: String
.open var className: String
.open val classList: DOMTokenList
.open var slot: String
.open val attributes: NamedNodeMap
.open val shadowRoot: ShadowRoot?
.open var scrollTop: Double
.open var scrollLeft: Double
.open val scrollWidth: Int
.open val scrollHeight: Int
.open val clientTop: Int
.open val clientLeft: Int
.open val clientWidth: Int
.open val clientHeight: Int
.open var innerHTML: String
.open var outerHTML: String

fun hasAttributes(): Boolean
fun getAttributeNames(): Array<String>
fun getAttribute(qualifiedName: String): String?
fun getAttributeNS(namespace: String?, localName: String): String?
fun setAttribute(qualifiedName: String, value: String)
fun setAttributeNS(namenspace: String, localName: String, value: String)
fun removeAttribute(qualifiedName: String)
fun removeAttributeNS(namespace: String, localName: String)
fun hasAttribute(qualifiedName: String): Boolean
fun hasAttributeNS(namenspace: String, localName: String)
fun hasAttributeNodeNS(namenspace: String, localName: String)
fun getAttributeNode(qualifiedName: String)
fun getAttributeNodeNS(namenspace: String, localName: String)
fun hasAttributeNodeNS(namenspace: String, localName: String)
fun getAttributeNodeNS(qualifiedName: String)
fun getAttributeNodeNS(attr: Attr)
fun hasAttributeNodeNS(qualifiedName: String)
fun hasAttributeNodeNS(attr: Attr)
fun removeAttributeNode(attr: Attr)
fun attachShadow(init: ShadowRootInit)
/**
 * Exposes the JavaScript [Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin
 */
public external open class Text(data: String = definedExternally) : CharacterData, Slotable, GeometryUtils {
    open val wholeText: String
    override val assignedSlot: HTMLSlotElement?
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    fun splitText(offset: Int): Text
    override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
    override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin
 */
public external open class CDATASection : Text {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin
 */
public external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {
    open val target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

/**
 * Exposes the JavaScript [Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin
 */
public external open class Comment(data: String = definedExternally) : CharacterData {
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
 */
public external open class Range {
    open val startContainer: Node
    open val startOffset: Int
    open val endContainer: Node
    open val endOffset: Int
    open val collapsed: Boolean
    open val commonAncestorContainer: Node
    fun setStart(node: Node, offset: Int)
    fun setEnd(node: Node, offset: Int)
    fun setStartBefore(node: Node)
    fun setStartAfter(node: Node)
    fun setEndBefore(node: Node)
    fun setEndAfter(node: Node)
    fun collapse(toStart: Boolean = definedExternally)
    fun selectNode(node: Node)
    fun selectNodeContents(node: Node)
    fun compareBoundaryPoints(how: Short, sourceRange: Range): Short
    fun deleteContents()
    fun extractContents(): DocumentFragment
    fun cloneContents(): DocumentFragment
    fun insertNode(node: Node)
    fun surroundContents(newParent: Node)
    fun cloneRange(): Range
    fun detach()
    fun isPointInRange(node: Node, offset: Int): Boolean
    fun comparePoint(node: Node, offset: Int): Short
    fun intersectsNode(node: Node: Node): Boolean
    fun getClientRects(): Array<DOMRect>
    fun getBoundingClientRect(): DOMRect
    fun createContextualFragment(fragment: String): DocumentFragment

    companion object {
        val START_TO_START: Short
        val START_TO_END: Short
        val END_TO_END: Short
        val END_TO_START: Short
    }
}

/**
 * Exposes the JavaScript [NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
    open val root: Node
    open val referenceNode: Node
    open val pointerBeforeReferenceNode: Boolean
    open val whatToShow: Int
    open val filter: NodeFilter?
    fun nextNode(): Node?
    fun previousNode(): Node?
    fun detach()
}

/**
 * Exposes the JavaScript [TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
    open val root: Node
    open val whatToShow: Int
    open val filter: NodeFilter?
    open var currentNode: Node
    fun parentNode(): Node?
    fun firstChild(): Node?
    fun lastChild(): Node?
    fun previousSibling(): Node?
    fun nextSibling(): Node?
    fun previousNode(): Node?
    fun nextNode(): Node?
}

/**
 * Exposes the JavaScript [NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
    fun acceptNode(node: Node): Short

    companion object {
        val FILTER_ACCEPT: Short
        val FILTER_REJECT: Short
        val FILTER_SKIP: Short
        val SHOW_ALL: Int
        val SHOW_ELEMENT: Int
        val SHOW_ATTRIBUTE: Int
        val SHOW_TEXT: Int
        val SHOW_CDATA_SECTION: Int
        val SHOW_ENTITY_REFERENCE: Int
        val SHOW_ENTITY: Int
        val SHOW_PROCESSING_INSTRUCTION: Int
        val SHOW_COMMENT: Int
    }
}
SHOW_DOCUMENT: Int
val SHOW_DOCUMENT_TYPE: Int

SHOW_DOCUMENT_FRAGMENT: Int
val SHOW_NOTATION: Int

/**
 * Exposes the
JavaScript [DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
*/
public external abstract class DOMTokenList : ItemArrayLike<String> {
    open var value: String
    fun contains(vararg tokens: String): Boolean
    fun add(vararg tokens: String): Boolean
    fun remove(vararg tokens: String): Boolean
    fun toggle(vararg tokens: String): Boolean
    fun supports(vararg tokens: String): Boolean
    override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()[index]

/**
 * Exposes the JavaScript
[DOMPointReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMPointReadOnly) to Kotlin
*/
public external open class DOMPointReadOnly(x: Double, y: Double, z: Double, w: Double) {
    open val x: Double
    open val y: Double
    open val z: Double
    open val w: Double
    fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint
}

/**
 * Exposes the JavaScript
[DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin
*/
public external open class DOMPoint : DOMPointReadOnly {
    constructor(point: DOMPointInit)
    constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)
    override var x: Double
    override var y: Double
    override var z: Double
    override var w: Double
}

/**
 * Exposes the JavaScript
[DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin
*/
public external interface DOMPointInit {
    var x: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var y: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var z: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var w: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["w"] = w
    return o
}

/**
 * Exposes the JavaScript
[DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin
*/
public external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {
    override var x: Double
    override var y: Double
    override var width: Double
    override var height: Double
}

/**
 * Exposes the JavaScript
[DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin
*/
public external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {
    open val x: Double
    open val y: Double
    open val width: Double
    open val height: Double
}

public external interface DOMRectList : ItemArrayLike<DOMRect> {
    override fun item(index: Int): DOMRect?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]

/**
 * Exposes the JavaScript
[DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin
*/
public external open class DOMQuad {
    constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)
    constructor(rect: DOMRectInit)
    open val p1: DOMPoint
    open val p2: DOMPoint
    open val p3: DOMPoint
    open val p4: DOMPoint
}
open val bounds: DOMRectReadOnly

public external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {

    open val a: Double
    open val b: Double
    open val c: Double
    open val d: Double
    open val e: Double
    open val f: Double
    open val m11: Double
    open val m12: Double
    open val m13: Double
    open val m14: Double
    open val m21: Double
    open val m22: Double
    open val m23: Double
    open val m24: Double
    open val m31: Double
    open val m32: Double
    open val m33: Double
    open val m34: Double
    open val m41: Double
    open val m42: Double
    open val m43: Double
    open val m44: Double
    open val is2D: Boolean
    open val isIdentity: Boolean
    fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun scaleNonUniform(scaleX: Double, scaleY: Double = definedExternally): DOMMatrix
    fun scaleNonUniform(scaleX: Double, scaleY: Double, scaleZ: Double = definedExternally): DOMMatrix
    fun multiply(other: DOMMatrix): DOMMatrix
    fun flipX(): DOMMatrix
    fun flipY(): DOMMatrix
    fun inverse(): DOMMatrix
    fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint
    fun toFloat32Array(): Float32Array
    fun toFloat64Array(): Float64Array
}

public external open class DOMMatrix() : DOMMatrixReadOnly {

    constructor(transformList: String)
    constructor(other: DOMMatrixReadOnly)
    constructor(array32: Float32Array)
    constructor(array64: Float64Array)
    constructor(numberSequence: Array<Double>)
    override var a: Double
    override var b: Double
    override var c: Double
    override var d: Double
    override var e: Double
    override var f: Double
    override var m11: Double
    override var m12: Double
    override var m13: Double
    override var m14: Double
    override var m21: Double
    override var m22: Double
    override var m23: Double
    override var m24: Double
    override var m31: Double
    override var m32: Double
    override var m33: Double
    override var m34: Double
    override var m41: Double
    override var m42: Double
    override var m43: Double
    override var m44: Double
    fun multiplySelf(other: DOMMatrix): DOMMatrix
    fun preMultiplySelf(other: DOMMatrix): DOMMatrix
    fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3dSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally): DOMMatrix
    fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVectorSelf(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngleSelf(x: Double, y: Double): DOMMatrix
    fun skewXSelf(sx: Double): DOMMatrix
    fun skewYSelf(sy: Double): DOMMatrix
    fun invertSelf(): DOMMatrix
    fun setMatrixValue(transformList: String): DOMMatrix
}

public external interface ScrollOptions {
    var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
    fun get(): definedExternally
    fun set(value): definedExternally
}

@kotlin.internal.InlineOnly
fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO, get(): definedExternally, set(value): definedExternally) = ScrollOptions(behavior, get(), set(value))
Open Source Used In DNAC Platform Arica  13417

```kotlin
val o = js("({})")
o["left"] = left
o["top"] = top
o["behavior"] = behavior
return o

fun addListener(listener: EventListener?)
fun addListener(listener: (Event) -> Unit?)
fun removeListener(listener: EventListener?)
fun removeListener(listener: (Event) -> Unit?): Unit

/**
 Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
    open val media: String
    open val matches: Boolean
    open var onchange: ((Event) -> dynamic)?
    fun addListener(listener: EventListener?)
    fun addListener(listener: (Event) -> Unit?)
    fun removeListener(listener: EventListener?)
    fun removeListener(listener: (Event) -> Unit?): Unit

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaQueryListEventInit : EventInit {
    var media: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var matches: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["media"] = media
    o["matches"] = matches
    return o
}

/**
 * Exposes the JavaScript [Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin
 */
public external abstract class Screen {
    open val availWidth: Int
    open val availHeight: Int
    open val width: Int
    open val height: Int
    open val colorDepth: Int
    open val pixelDepth: Int
}

/**
 * Exposes the JavaScript [CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin
 */
public external abstract class CaretPosition {
    open val offsetNode: Node
    open val offset: Int
    fun getClientRect(): DOMRect?
}

public external interface ScrollIntoViewOptions : ScrollOptions {
    var block: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
        get() = definedExternally
        set(value) = definedExternally
    var inline: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["block"] = block
    o["inline"] = inline
    return o
}

public external interface BoxQuadOptions {
    var box: CSSBoxType? /* = CSSBoxType.BORDER */
        get() = definedExternally
        set(value) = definedExternally
    var relativeTo: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun BoxQuadOptions(box: CSSBoxType? = CSSBoxType.BORDER, relativeTo: dynamic = undefined): BoxQuadOptions {
    val o = js("({})")
    o["box"] = box
    o["relativeTo"] = relativeTo
    return o
}

public external interface ConvertCoordinateOptions {
    var fromBox: CSSBoxType? /* = CSSBoxType.BORDER */
        get() = definedExternally
        set(value) = definedExternally
    var toBox: CSSBoxType?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConvertCoordinateOptions(fromBox: CSSBoxType? = CSSBoxType.BORDER, toBox: CSSBoxType?: CSSBoxType.BORDER): ConvertCoordinateOptions {
    val o = js("({})")
    o["fromBox"] = fromBox
    o["toBox"] = toBox
    return o
}

public external interface GeometryUtils {
    fun getBoxQuads(options: BoxQuadOptions = definedExternally): DOMQuad
    fun convertQuadFromNode(quad: DOMQuad, dynamic: from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
    fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions = definedExternally): DOMQuad
    fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions =
```
definedExternally): DOMPoint\n}\n\n/**\n * Exposes the JavaScript
[Touch](https://developer.mozilla.org/en/docs/Web/API/Touch) to Kotlin\n */\npublic external abstract class Touch
{\n open val identifier: Int\n open val target: EventTarget\n open val screenX: Int\n open val screenY: Int\n
open val clientX: Int\n open val clientY: Int\n open val pageX: Int\n open val pageY: Int\n open val region:
String?\n}\n\npublic external abstract class TouchList : ItemArrayLike<Touch> {\n override fun item(index: Int):
Touch?\n}\n\n@kotlin.internal.InlineOnly\npublic inline operator fun TouchList.get(index: Int): Touch? =
asDynamic()[index]\n\npublic external open class TouchEvent : UIEvent {\n open val touches: TouchList\n
open val targetTouches: TouchList\n open val changedTouches: TouchList\n open val altKey: Boolean\n open
val metaKey: Boolean\n open val ctrlKey: Boolean\n open val shiftKey: Boolean\n\n companion object {\n
val NONE: Short\n
val CAPTURING_PHASE: Short\n
val AT_TARGET: Short\n
val
BUBBLING_PHASE: Short\n }\n}\n\n/**\n * Exposes the JavaScript
[Image](https://developer.mozilla.org/en/docs/Web/API/Image) to Kotlin\n */\npublic external open class
Image(width: Int = definedExternally, height: Int = definedExternally) : HTMLImageElement {\n override var
onabort: ((Event) -> dynamic)?\n override var onblur: ((FocusEvent) -> dynamic)?\n override var oncancel:
((Event) -> dynamic)?\n override var oncanplay: ((Event) -> dynamic)?\n override var oncanplaythrough:
((Event) -> dynamic)?\n override var onchange: ((Event) -> dynamic)?\n override var onclick: ((MouseEvent) ->
dynamic)?\n override var onclose: ((Event) -> dynamic)?\n override var oncontextmenu: ((MouseEvent) ->
dynamic)?\n override var oncuechange: ((Event) -> dynamic)?\n override var ondblclick: ((MouseEvent) ->
dynamic)?\n override var ondrag: ((DragEvent) -> dynamic)?\n override var ondragend: ((DragEvent) ->
dynamic)?\n override var ondragenter: ((DragEvent) -> dynamic)?\n override var ondragexit: ((DragEvent) ->
dynamic)?\n override var ondragleave: ((DragEvent) -> dynamic)?\n override var ondragover: ((DragEvent) ->
dynamic)?\n override var ondragstart: ((DragEvent) -> dynamic)?\n override var ondrop: ((DragEvent) ->
dynamic)?\n override var ondurationchange: ((Event) -> dynamic)?\n override var onemptied: ((Event) ->
dynamic)?\n override var onended: ((Event) -> dynamic)?\n override var onerror: ((dynamic, String, Int, Int,
Any?) -> dynamic)?\n override var onfocus: ((FocusEvent) -> dynamic)?\n override var oninput: ((InputEvent) > dynamic)?\n override var oninvalid: ((Event) -> dynamic)?\n override var onkeydown: ((KeyboardEvent) ->
dynamic)?\n override var onkeypress: ((KeyboardEvent) -> dynamic)?\n override var onkeyup:
((KeyboardEvent) -> dynamic)?\n override var onload: ((Event) -> dynamic)?\n override var onloadeddata:
((Event) -> dynamic)?\n override var onloadedmetadata: ((Event) -> dynamic)?\n override var onloadend:
((Event) -> dynamic)?\n override var onloadstart: ((ProgressEvent) -> dynamic)?\n override var onmousedown:
((MouseEvent) -> dynamic)?\n override var onmouseenter: ((MouseEvent) -> dynamic)?\n override var
onmouseleave: ((MouseEvent) -> dynamic)?\n override var onmousemove: ((MouseEvent) -> dynamic)?\n
override var onmouseout: ((MouseEvent) -> dynamic)?\n override var onmouseover: ((MouseEvent) ->
dynamic)?\n override var onmouseup: ((MouseEvent) -> dynamic)?\n override var onwheel: ((WheelEvent) ->
dynamic)?\n override var onpause: ((Event) -> dynamic)?\n override var onplay: ((Event) -> dynamic)?\n
override var onplaying: ((Event) -> dynamic)?\n override var onprogress: ((ProgressEvent) -> dynamic)?\n
override var onratechange: ((Event) -> dynamic)?\n override var onreset: ((Event) -> dynamic)?\n override var
onresize: ((Event) -> dynamic)?\n override var onscroll: ((Event) -> dynamic)?\n override var onseeked:
((Event) -> dynamic)?\n override var onseeking: ((Event) -> dynamic)?\n override var onselect: ((Event) ->
dynamic)?\n override var onshow: ((Event) -> dynamic)?\n override var onstalled: ((Event) -> dynamic)?\n
override var onsubmit: ((Event) -> dynamic)?\n override var onsuspend: ((Event) -> dynamic)?\n override var
ontimeupdate: ((Event) -> dynamic)?\n override var ontoggle: ((Event) -> dynamic)?\n override var
onvolumechange: ((Event) -> dynamic)?\n override var onwaiting: ((Event) -> dynamic)?\n override var
ongotpointercapture: ((PointerEvent) -> dynamic)?\n override var onlostpointercapture: ((PointerEvent) ->
dynamic)?\n override var onpointerdown: ((PointerEvent) -> dynamic)?\n override var onpointermove:
((PointerEvent) -> dynamic)?\n override var onpointerup: ((PointerEvent) -> dynamic)?\n override var
onpointercancel: ((PointerEvent) -> dynamic)?\n override var onpointerover: ((PointerEvent) -> dynamic)?\n
override var onpointerout: ((PointerEvent) -> dynamic)?\n override var onpointerenter: ((PointerEvent) ->

Open Source Used In DNAC Platform Arica 13418


override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy:
((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste:
((ClipboardEvent) -> dynamic)?
override var onpaste:
((ClipboardEvent) -> dynamic)?
override var contentEditable: String?
override val isContentEditable: Boolean
override val style: CSSStyleDeclaration
override val children: HTMLCollection
override val firstElementChild: Element?
override val lastElementChild: Element?
override val childElementCount: Int
override val previousElementSibling: Element?
override val nextElementSibling: Element?
override val assignedSlot: HTMLSlotElement?
override fun prepend(vararg nodes: dynamic)
override fun append(vararg nodes: dynamic)
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad?

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external open class Audio(src: String = definedExternally) : HTMLAudioElement {
override var onprogress: ((ProgressEvent) -> dynamic)?
override var onratechange: ((Event) -> dynamic)?
override var onreset: ((Event) -> dynamic)?
override var onsize: ((Event) -> dynamic)?
override var onscroll: ((Event) -> dynamic)?
override var onsecked: ((Event) -> dynamic)?
override var onseeking: ((Event) -> dynamic)?
override var onselect: ((Event) -> dynamic)?
override var onshow: ((Event) -> dynamic)?
override var onstalled: ((Event) -> dynamic)?
override var onsubmit: ((Event) -> dynamic)?
override var onsuspend: ((Event) -> dynamic)?
override var ontimeupdate: ((Event) -> dynamic)?
override var ontoggle: ((Event) -> dynamic)?
override var onvolumechange: ((Event) -> dynamic)?
override var onwaiting: ((Event) -> dynamic)?
override var ongpointercapture: ((PointerEvent) -> dynamic)?
override var onpointermove: ((PointerEvent) -> dynamic)?
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerover: ((PointerEvent) -> dynamic)?
override var onpointerout: ((PointerEvent) -> dynamic)?
override var onpointerenter: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy: ((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste: ((ClipboardEvent) -> dynamic)?
override var contentEditable: String
override val isContentEditable: Boolean
override val style: CSSStyleDeclaration
override val children: HTMLCollection
override val firstElementChild: Element?
override val lastElementChild: Element?
override val childElementCount: Int
override val previousElementSibling: Element?
override val nextElementSibling: Element?
override val assignedSlot: HTMLSlotElement?
override fun prepend(vararg nodes: Dynamic): Dynamic
override fun append(vararg nodes: Dynamic): Dynamic
override fun querySelector(selectors: String): Element?
override fun querySelectorAll(selectors: String): NodeList
override fun before(vararg nodes: Dynamic): Dynamic
override fun after(vararg nodes: Dynamic): Dynamic
override fun replaceWith(vararg nodes: Dynamic): Dynamic
override fun querySelectorAll(selectors: String): NodeList
override fun querySelector(selectors: String): Element?
override fun remove(): Dynamic
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: Dynamic, from: Dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: Dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertPointFromNode(point: DOMPointInit, from: Dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
override var NETWORK_EMPTY: Short
override var NETWORK_IDLE: Short
override var NETWORK_LOADING: Short
override var NETWORK_NO_SOURCE: Short
override var HAVE_NOTHING: Short
override var HAVE_METADATA: Short
override var HAVE_CURRENT_DATA: Short
override var HAVE_FUTURE_DATA: Short
override var HAVE_ENOUGH_DATA: Short
override var ELEMENT_NODE: Short
override var ATTRIBUTE_NODE: Short
override var TEXT_NODE: Short
override var CDATA_SECTION_NODE: Short
override var ENTITYREFERENCE_NODE: Short
override var ENTITY_NODE: Short
override var PROCESSING_INSTRUCTION_NODE: Short
override var COMMENT_NODE: Short
override var DOCUMENT_NODE: Short
override var DOCUMENT_TYPE_NODE: Short
override var DOCUMENT_FRAGMENT_NODE: Short
override var DOCUMENT_POSITION_DISCONNECTED: Short
override var DOCUMENT_POSITION_FOLLOWING: Short
override var DOCUMENT_POSITION_CONTAINS: Short
override var DOCUMENT_POSITION_CONTAINED_BY: Short
override var DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
override var onabort: ((Event) -> dynamic)?
override var onblur: ((FocusEvent) -> dynamic)?
override var oncancel: ((Event) -> dynamic)?
override oncanplay: ((Event) -> dynamic)?
override oncanplaythrough: ((Event) -> dynamic)?
override onchange: ((Event) -> dynamic)?
override onclick: ((MouseEvent) -> dynamic)?
override onclose: ((Event) -> dynamic)?
override oncontextmenu: ((MouseEvent) -> dynamic)?
override oncontextchange: ((Event) -> dynamic)?
"select".asDynamic().unsafeCast<SelectionMode>().

public inline val SelectionMode.Companion.START: SelectionMode get() = "start".asDynamic().unsafeCast<SelectionMode>().

public inline val SelectionMode.Companion.PRESERVE: SelectionMode get() = "preserve".asDynamic().unsafeCast<SelectionMode>().

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasFillRule {
    companion object
}

public inline val CanvasFillRule.Companion.NONZERO: CanvasFillRule get() = "nonzero".asDynamic().unsafeCast<CanvasFillRule>().

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineCap {
    companion object
}

public inline val CanvasLineCap.Companion.BUTT: CanvasLineCap get() = "butt".asDynamic().unsafeCast<CanvasLineCap>().

public inline val CanvasLineCap.Companion.SQUARE: CanvasLineCap get() = "square".asDynamic().unsafeCast<CanvasLineCap>().

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasLineJoin {
    companion object
}

public inline val CanvasLineJoin.Companion.ROUND: CanvasLineJoin get() = "round".asDynamic().unsafeCast<CanvasLineJoin>().

public inline val CanvasLineJoin.Companion.BEVEL: CanvasLineJoin get() = "bevel".asDynamic().unsafeCast<CanvasLineJoin>().

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextAlign {
    companion object
}

public inline val CanvasTextAlign.Companion.START: CanvasTextAlign get() = "start".asDynamic().unsafeCast<CanvasTextAlign>().

public inline val CanvasTextAlign.Companion.END: CanvasTextAlign get() = "end".asDynamic().unsafeCast<CanvasTextAlign>().

public inline val CanvasTextAlign.Companion.LEFT: CanvasTextAlign get() = "left".asDynamic().unsafeCast<CanvasTextAlign>().

public inline val CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() = "center".asDynamic().unsafeCast<CanvasTextAlign>().

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextBaseline {
    companion object
}

public inline val CanvasTextBaseline.Companion.TOP: CanvasTextBaseline get() = "top".asDynamic().unsafeCast<CanvasTextBaseline>().

public inline val CanvasTextBaseline.Companion.MIDDLE: CanvasTextBaseline get() = "middle".asDynamic().unsafeCast<CanvasTextBaseline>().

public inline val CanvasTextBaseline.Companion.ALPHABETIC: CanvasTextBaseline get() = "alphabetic".asDynamic().unsafeCast<CanvasTextBaseline>().

public inline val CanvasTextBaseline.Companion.IDEOGRAPHIC: CanvasTextBaseline get() =...
CanvasTextBaseline.Companion.BOTTOM: CanvasTextBaseline get() = "bottom".asDynamic().unsafeCast<CanvasTextBaseline>()

CanvasDirection.Companion.LTR: CanvasDirection get() = "ltr".asDynamic().unsafeCast<CanvasDirection>()

CanvasDirection.Companion.RTL: CanvasDirection get() = "rtl".asDynamic().unsafeCast<CanvasDirection>()

CanvasDirection.Companion.INHERIT: CanvasDirection get() = "inherit".asDynamic().unsafeCast<CanvasDirection>()

ScrollRestoration.Companion.AUTO: ScrollRestoration get() = "auto".asDynamic().unsafeCast<ScrollRestoration>()

ScrollRestoration.Companion.MANUAL: ScrollRestoration get() = "manual".asDynamic().unsafeCast<ScrollRestoration>()


ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() = "none".asDynamic().unsafeCast<ColorSpaceConversion>()

ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() = "default".asDynamic().unsafeCast<ColorSpaceConversion>()

ResizeQuality.Companion.PIXELATED: ResizeQuality get() = "pixelated".asDynamic().unsafeCast<ResizeQuality>()

ResizeQuality.Companion.LOW: ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()

ResizeQuality.Companion.MEDIUM: ResizeQuality get() = "medium".asDynamic().unsafeCast<ResizeQuality>()

ResizeQuality.Companion.HIGH: ResizeQuality get() = "high".asDynamic().unsafeCast<ResizeQuality>()

BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()

BinaryType.Companion.ARRAYBUFFER: BinaryType get() = "arraybuffer".asDynamic().unsafeCast<BinaryType>()

WorkerType.Companion.CLASSIC: WorkerType get() = "classic".asDynamic().unsafeCast<WorkerType>()

WorkerType.Companion.MODULE: WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()

ShadowRootMode.Companion.OPEN: ShadowRootMode get() = "open".asDynamic().unsafeCast<ShadowRootMode>()
public inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() = "auto".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() = "instant".asDynamic().unsafeCast<ScrollBehavior>()

public inline val ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() = "smooth".asDynamic().unsafeCast<ScrollBehavior>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollLogicalPosition {
    companion object
}


public inline val ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() = "center".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() = "end".asDynamic().unsafeCast<ScrollLogicalPosition>()

public inline val ScrollLogicalPosition.Companion.NEAREST: ScrollLogicalPosition get() = "nearest".asDynamic().unsafeCast<ScrollLogicalPosition>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CSSBoxType {
    companion object
}

public inline val CSSBoxType.Companion.MARGIN: CSSBoxType get() = "margin".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.BORDER: CSSBoxType get() = "border".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.PADDING: CSSBoxType get() = "padding".asDynamic().unsafeCast<CSSBoxType>()

public inline val CSSBoxType.Companion.CONTENT: CSSBoxType get() = "content".asDynamic().unsafeCast<CSSBoxType>()
get() = definedExternally
set(value) = definedExternally
var headers: dynamic
get() = definedExternally
set(value) = definedExternally
var body: dynamic
get() = definedExternally
set(value) = definedExternally
var referer: String?
get() = definedExternally
set(value) = definedExternally
var refererPolicy: dynamic
get() = definedExternally
set(value) = definedExternally
var mode: RequestMode?
get() = definedExternally
set(value) = definedExternally
var credentials: RequestCredentials?
get() = definedExternally
set(value) = definedExternally
var cache: RequestCache?
get() = definedExternally
set(value) = definedExternally
var redirect: RequestRedirect?
get() = definedExternally
set(value) = definedExternally
var integrity: String?
get() = definedExternally
set(value) = definedExternally
var keepalive: Boolean?
get() = definedExternally
set(value) = definedExternally
var window: Any?
get() = definedExternally
set(value) = definedExternally
@kotlin.internal.InlineOnly
public inline fun RequestInit(method: String? = undefined,
    val o = js("({})")
    o["method"] = method
    o["headers"] = headers
    o["body"] = body
c}
Open Source Used In DNAC Platform Arica 13428

```java
RequestCache get() = \"no-store\".asDynamic().unsafeCast<RequestCache>(\n
public inline val RequestCache.Companion.RELOAD: RequestCache get() = \"reload\".asDynamic().unsafeCast<RequestCache>(\n
public inline val RequestCache.Companion.NO_CACHE: RequestCache get() = \"no-cache\".asDynamic().unsafeCast<RequestCache>(\n
public inline val RequestCache.Companion.FORCE_CACHE: RequestCache get() = \"force-cache\".asDynamic().unsafeCast<RequestCache>(\n
public inline val RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = \"only-if-cached\".asDynamic().unsafeCast<RequestCache>(\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestRedirect {
    companion object
}

public inline val RequestRedirect.Companion.FOLLOW: RequestRedirect get() = \"follow\".asDynamic().unsafeCast<RequestRedirect>(\n
public inline val RequestRedirect.Companion.ERROR: RequestRedirect get() = \"error\".asDynamic().unsafeCast<RequestRedirect>(\n
public inline val RequestRedirect.Companion.MANUAL: RequestRedirect get() = \"manual\".asDynamic().unsafeCast<RequestRedirect>(\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResponseType {
    companion object
}

public inline val ResponseType.Companion.BASIC: ResponseType get() = \"basic\".asDynamic().unsafeCast<ResponseType>(\n
public inline val ResponseType.Companion.CORS: ResponseType get() = \"cors\".asDynamic().unsafeCast<ResponseType>(\n
public inline val ResponseType.Companion.DEFAULT: ResponseType get() = \"default\".asDynamic().unsafeCast<ResponseType>(\n
public inline val ResponseType.Companion.OPAQUE: ResponseType get() = \"opaque\".asDynamic().unsafeCast<ResponseType>(\n
public inline val ResponseType.Companion.OPAQUEREDIRECT: ResponseType get() = \"opaqueredirect\".asDynamic().unsafeCast<ResponseType>(\n
*/
```

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.dom.mediacapture

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.parsing.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin
 */
public external open class MediaStream() : EventTarget {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}

/**
 * Exposes the JavaScript [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin
 */
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open var muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop()
    fun getCapabilities(): MediaTrackCapabilities
    fun getConstraints(): MediaTrackConstraints
    fun getSettings(): MediaTrackSettings
    fun applyConstraints(constraints:
MediaTrackConstraints = definedExternally): Promise<Unit?>

/**
 * Exposes the JavaScript
 * [MediaTrackSupportedConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSupportedConstraints)
 * to Kotlin
 */

public external interface MediaTrackSupportedConstraints {
    var width: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var height: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var volume: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var latency: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var groupId: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaTrackSupportedConstraints(width: Boolean? = true, height: Boolean? = true, aspectRatio: Boolean? = true,
    frameRate: Boolean? = true, facingMode: Boolean? = true, resizeMode: Boolean? = true, volume: Boolean? = true,
    sampleRate: Boolean? = true, sampleSize: Boolean? = true, echoCancellation: Boolean? = true, autoGainControl: Boolean? = true,
    noiseSuppression: Boolean? = true, latency: Boolean? = true, channelCount: Boolean? = true, deviceId: Boolean? = true,
    groupId: Boolean? = true): MediaTrackSupportedConstraints {
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    return o
}

public external interface MediaTrackCapabilities {
    var width: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var height: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var volume: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: Array<Boolean>?
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: Array<Boolean>?
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: Array<Boolean>?
        get() = definedExternally
        set(value) = definedExternally
    var latency: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly

Open Source Used In DNAC Platform Arica  13429
undefined, volume: DoubleRange? = undefined, sampleRate: ULongRange? = undefined, sampleSize:
ULongRange? = undefined, echoCancellation: Array<Boolean>? = undefined, latency: DoubleRange? = undefined, channelCount:
ULongRange? = undefined, deviceId: String? = undefined, groupId: String? = undefined): MediaTrackCapabilities
{
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript
[MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin
 */
public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced:
Array<MediaTrackConstraintSet>?
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraints(advanced:
Array<MediaTrackConstraintSet>? = undefined, width: dynamic = undefined, height: dynamic = undefined,
aspectRatio: dynamic = undefined, frameRate: dynamic = undefined, facingMode: dynamic = undefined,
resizeMode: dynamic = undefined, sampleRate: dynamic = undefined, sampleSize: dynamic = undefined,
echoCancellation: dynamic = undefined, autoGainControl: dynamic = undefined,
noiseSuppression: dynamic = undefined, latency: dynamic = undefined, channelCount: dynamic = undefined,
deviceId: dynamic = undefined, groupId: dynamic = undefined): MediaTrackConstraints
{
    val o = js("({})")
    o["advanced"] = advanced
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

public external interface MediaTrackConstraintSet {
    var width: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var height: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var latency: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var groupId: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun MediaTrackConstraintSet(width: dynamic = undefined, height: dynamic = undefined,
aspectRatio: dynamic = undefined, frameRate: dynamic = undefined, facingMode: dynamic = undefined,
resizeMode: dynamic = undefined, deviceType: String? = undefined, groupId: String? = undefined):
MediaTrackConstraintSet
{
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}
Open Source Used In DNAC Platform Arica 13431

facingMode: String
    o["resizeModel"] = resizeModel
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
return o
}

/**
 * Exposes the JavaScript
 **/[MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Double?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var latency: Double?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeModel"] = resizeModel
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
return o
}

/**
 * Exposes the JavaScript
 **/[MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin
 */
public external open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) : Event {
    open val track: MediaStreamTrack

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaStreamTrackEventInit : EventInit {
    var track: MediaStreamTrack?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["track"] = track
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
return o
}

public external open class OverconstrainedErrorEvent(type: String, eventInitDict: OverconstrainedErrorEventInit) : Event {
    var error: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
OverconstrainedErrorEventInit {\n    val o = js("({})")\n    o["error"] = error\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n}\n
* Exposes the JavaScript MediaDevices to Kotlin

fun getSupportedConstraints(): MediaDeviceInfo {
    val o = js("({})")\n    o["error"] = error\n    o["bubbles"] = bubbles\n    o["cancelable"] = cancelable\n    o["composed"] = composed\n    return o\n}\n
/**
 * Exposes the JavaScript MediaDevices to Kotlin
 */
public external abstract class MediaDevices : EventTarget {
    open var ondevicechange: (Event) -> dynamic?\n    fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>\n    fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>\n}

/**
 * Exposes the JavaScript MediaDeviceInfo to Kotlin
 */
public external abstract class MediaDeviceInfo {
    open var deviceId: String?\n    open var kind: MediaDeviceKind?\n    open var label: String?\n    open var groupId: String?\n    fun toJSON(): dynamic\n}

public external abstract class InputDeviceInfo : MediaDeviceInfo {
    fun getCapabilities(): MediaTrackCapabilities\n}

/**
 * Exposes the JavaScript MediaStreamConstraints to Kotlin
 */
public external interface MediaStreamConstraints {
    var video: dynamic /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n    var audio: dynamic /* = false */\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
public inline fun MediaStreamConstraints(video: dynamic = false, audio: dynamic = false): MediaStreamConstraints {
    val o = js("({})")\n    o["video"] = video\n    o["audio"] = audio\n    return o\n}\n
public external interface ConstrainablePattern {
    var onoverconstrained: (Event) -> dynamic?\n    fun getCapabilities(): Capabilities\n    fun getConstraints(): Constraints\n    fun getSettings(): Settings?\n    fun applyConstraints(constraints: Constraints = definedExternally): Promise<Unit>\n}

/**
 * Exposes the JavaScript DoubleRange to Kotlin
 */
public external interface DoubleRange {
    var max: Double?\n        get() = definedExternally\n        set(value) = definedExternally\n    var min: Double?\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
    val o = js("({})")\n    o["max"] = max\n    o["min"] = min\n    return o\n}\n
public external interface ConstrainDoubleRange : DoubleRange {
    var exact: Double?\n        get() = definedExternally\n        set(value) = definedExternally\n    var ideal: Double?\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
    val o = js("({})")\n    o["exact"] = exact\n    o["ideal"] = ideal\n    o["max"] = max\n    o["min"] = min\n    return o\n}\n
public external interface ULongRange {
    var max: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n    var min: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
public inline fun ULongRange(max: Int? = undefined, min: Int? = undefined): ULongRange {
    val o = js("({})")\n    o["max"] = max\n    o["min"] = min\n    return o\n}\n
public external interface ConstrainULongRange : ULongRange {
    var exact: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n    var ideal: Int?\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
    val o = js("({})")\n    o["exact"] = exact\n    o["ideal"] = ideal\n    o["max"] = max\n    o["min"] = min\n    return o\n}\n
public external interface ConstrainBooleanParameters {
    var exact: Boolean?\n        get() = definedExternally\n        set(value) = definedExternally\n    var ideal: Boolean?\n        get() = definedExternally\n        set(value) = definedExternally\n}

@kotlin.internal.InlineOnly
    val o = js("({})")\n    o["exact"] = exact\n    o["ideal"] = ideal\n    o["max"] = max\n    o["min"] = min\n    return o\n}\n
* Exposes the JavaScript MediaDevices to Kotlin

fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>
ConstrainDOMStringParameters

```kotlin
public external interface ConstrainDOMStringParameters {
    var exact: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var ideal: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConstrainDOMStringParameters(exact: dynamic = undefined, ideal: dynamic = undefined): ConstrainDOMStringParameters {
    val o = js("{}")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}

public external interface Capabilities

@kotlin.internal.InlineOnly
public inline fun Capabilities(): Capabilities {
    val o = js("{}")
    return o
}

public external interface Settings

@kotlin.internal.InlineOnly
public inline fun Settings(): Settings {
    val o = js("{}")
    return o
}

public external interface ConstraintSet

@kotlin.internal.InlineOnly
public inline fun ConstraintSet(): ConstraintSet {
    val o = js("{}")
    return o
}

public external interface Constraints : ConstraintSet {
    var advanced: Array<ConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints {
    val o = js("{}")
    o["advanced"] = advanced
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaStreamTrackState {
    companion object
}

public inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState get() = "live".asDynamic().unsafeCast<MediaStreamTrackState>()

public inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState get() = "ended".asDynamic().unsafeCast<MediaStreamTrackState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoFacingModeEnum {
    companion object
}

public inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum get() = "user".asDynamic().unsafeCast<VideoFacingModeEnum>()


public inline val VideoFacingModeEnum.Companion.LEFT: VideoFacingModeEnum get() = "left".asDynamic().unsafeCast<VideoFacingModeEnum>()

public inline val VideoFacingModeEnum.Companion.RIGHT: VideoFacingModeEnum get() = "right".asDynamic().unsafeCast<VideoFacingModeEnum>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoResizeModeEnum {
    companion object
}

public inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = "none".asDynamic().unsafeCast<VideoResizeModeEnum>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.VIDEOINPUT: MediaDeviceKind get() = "videoinput".asDynamic().unsafeCast<MediaDeviceKind>()
```

---

Open Source Used In DNAC Platform Arica  13433
import org.w3c.dom.mediacapture.
import org.w3c.dom.parsing.
import org.w3c.dom.svg.
import org.w3c.dom.url.
import org.w3c.fetch.
import org.w3c.files.
import org.w3c.notifications.
import org.w3c.performance.
import org.w3c.workers.
import org.w3c.xhr.

public external interface PointerEventInit : MouseEventInit {
    var pointerId: Int? /* = 0 */
        get() = definedExternally
    set(value) = definedExternally
    var width: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var height: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var pressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tangentialPressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tiltX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var tiltY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var twist: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var pointerType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isPrimary: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var screenX: Int
        get() = definedExternally
        set(value) = definedExternally
    var screenY: Int
        get() = definedExternally
        set(value) = definedExternally
    var clientX: Int
        get() = definedExternally
        set(value) = definedExternally
    var clientY: Int
        get() = definedExternally
        set(value) = definedExternally
    var button: Short
        get() = definedExternally
        set(value) = definedExternally
    var buttons: Short
        get() = definedExternally
        set(value) = definedExternally
    var relatedTarget: EventTarget
        get() = definedExternally
        set(value) = definedExternally
    var region: String
        get() = definedExternally
        set(value) = definedExternally
    var ctrlKey: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var shiftKey: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var altKey: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var metaKey: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var view: Window
        get() = definedExternally
        set(value) = definedExternally
    var detail: Int
        get() = definedExternally
        set(value) = definedExternally
    var bubbles: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var cancelable: Boolean
        get() = definedExternally
        set(value) = definedExternally
    var composed: Boolean
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["pointerId"] = pointerId
    o["width"] = width
    o["height"] = height
    o["pressure"] = pressure
    o["tangentialPressure"] = tangentialPressure
    o["tiltX"] = tiltX
    o["tiltY"] = tiltY
    o["twist"] = twist
    o["pointerType"] = pointerType
    o["isPrimary"] = isPrimary
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["Cancelable"] = cancelable
    o["Composed"] = composed
    o["return"] = return
    o["\n"] = 
  * Exposes the JavaScript
  [PointerEvent](https://developer.mozilla.org/en/docs/Web/API/PointerEvent) to Kotlin
  * `@public` external open class PointerEvent(type: String, eventInitDict: PointerEventInit = definedExternally): MouseEvent
  * `type` of pointerId: Int
  * `open` val width: Double
  * `open` val height: Double
  * `open` val pressure: Float
  * `open` val tangentialPressure: Float
  * `open` val tiltX: Int
  * `open` val tiltY: Int
  * `open` val twist: Int
  * `open` val pointerType: String
  * `open` val isPrimary: Boolean

* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
* See GitHub.com/kotlin/dukat for details
* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
* See GitHub.com/kotlin/dukat for details
* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
* See GitHub.com/kotlin/dukat for details
org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin
 */
public external abstract class SVGElement : Element, ElementCSSInlineStyle, GlobalEventHandlers,
SVGElementInstance {
    open val dataset: DOMStringMap
    open val ownerSVGElement: SVGSVGElement?
    open val viewportElement: SVGElement?
    open var tabIndex: Int
    fun focus()
    fun blur()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

public external interface SVGBoundingBoxOptions {
    var fill: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
    var stroke: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var markers: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var clipped: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["fill"] = fill
    o["stroke"] = stroke
    o["markers"] = markers
    o["clipped"] = clipped
    return o
}

/**
 * Exposes the JavaScript [SVGGraphicsElement](https://developer.mozilla.org/en/docs/Web/API/SVGGraphicsElement) to Kotlin
 */
public external abstract class SVGGraphicsElement : SVGElement, SVGTests {
    open val transform: SVGAnimatedTransformList
    fun getBBox(options: SVGBoundingBoxOptions = definedExternally): DOMRect
    fun getCTM(): DOMMatrix?
    fun getScreenCTM(): DOMMatrix?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin
 */
public external abstract class SVGGeometryElement : SVGGraphicsElement {
    open val pathLength: SVGAnimatedNumber
    fun isPointInFill(point: DOMPoint): Boolean
    fun isPointInStroke(point: DOMPoint): Boolean
    fun getTotalLength(): Float
    fun getPointAtLength(distance: Float): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGText](https://developer.mozilla.org/en/docs/Web/API/SVGText) to Kotlin
 */
public external abstract class SVGText : SVGElement, SVGUtilities {
    fun appendText(text: String)
    fun replaceText(text: String)
    fun removeText()
[SVGNumber](https://developer.mozilla.org/en/docs/Web/API/SVGNumber) to Kotlin */\n
```kotlin
public external abstract class SVGNumber {
    open var value: Float
}
```

/**
 * Exposes the JavaScript [SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin */

```kotlin
public external abstract class SVGLength {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    fun convertToSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun newValueSpecifiedUnits(unitType: Short, value: Float)
    fun getValueInSpecifiedUnits(): Float
    fun asString(): String
    companion object {
        val SVG_LENGTHTYPE_UNKNOWN: Short
        val SVG_LENGTHTYPE_NUMBER: Short
        val SVG_LENGTHTYPE_PERCENTAGE: Short
        val SVG_LENGTHTYPE_EM: Short
        val SVG_LENGTHTYPE_EX: Short
        val SVG_LENGTHTYPE_MM: Short
        val SVG_LENGTHTYPE_IN: Short
        val SVG_LENGTHTYPE_PT: Short
        val SVG_LENGTHTYPE_PC: Short
    }
}
```

/**
 * Exposes the JavaScript [SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin */

```kotlin
public external abstract class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    fun convertToSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun newValueSpecifiedUnits(unitType: Short, value: Float)
    fun getValueInSpecifiedUnits(): Float
    fun asString(): String
    companion object {
        val SVG_ANGLETYPE_UNKNOWN: Short
        val SVG_ANGLETYPE_UNSPECIFIED: Short
        val SVG_ANGLETYPE_DEG: Short
        val SVG_ANGLETYPE_RAD: Short
        val SVG_ANGLETYPE_GRAD: Short
    }
}
```

```kotlin
public external abstract class SVGNameList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: dynamic): dynamic
    fun insertItemBefore(newItem: dynamic, index: Int): dynamic
    fun replaceItem(newItem: dynamic, index: Int): dynamic
    fun removeItem(index: Int): dynamic
    fun appendItem(newItem: dynamic): dynamic
    fun getItem(index: Int): dynamic
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }
```

/**
 * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin */

```kotlin
public external abstract class SVGNumberList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGNumber): SVGNumber
    fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber
    fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber
    fun removeItem(index: Int): SVGNumber
    fun appendItem(newItem: SVGNumber): SVGNumber
    fun getItem(index: Int): SVGNumber
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }
```

/**
 * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin */

```kotlin
public external abstract class SVGLengthList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGLength): SVGLength
    fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength
    fun replaceItem(newItem: SVGLength, index: Int): SVGLength
    fun removeItem(index: Int): SVGLength
    fun appendItem(newItem: SVGLength): SVGLength
    fun getItem(index: Int): SVGLength
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength) { asDynamic()[index] = newItem }
```

/**
 * Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin */

```kotlin
public external abstract class SVGAnimatedBoolean {
    open var baseVal: Boolean
    open val animVal: Boolean
}
```

/**
 * Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin */

```kotlin
public external abstract class SVGAnimatedEnumeration {
    open var baseVal: Short
    open val animVal: Short
}
```
[SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin

*An external abstract class SVGAnimatedInteger

```kotlin
    open var baseVal: Int
    open val animVal: Int
```

Exposes the JavaScript

[SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin

*An external abstract class SVGAnimatedNumber

```kotlin
    open var baseVal: Float
    open val animVal: Float
```

Exposes the JavaScript

[SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin

*An external abstract class SVGAnimatedLength

```kotlin
    open val baseVal: SVGLength
    open val animVal: SVGLength
```

Exposes the JavaScript

[SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin

*An external abstract class SVGAnimatedAngle

```kotlin
    open val baseVal: SVGAngle
    open val animVal: SVGAngle
```

Exposes the JavaScript

[SVGAnimatedString](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedString) to Kotlin

*An external abstract class SVGAnimatedString

```kotlin
    open var baseVal: String
    open val animVal: String
```

Exposes the JavaScript

[SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect) to Kotlin

*An external abstract class SVGAnimatedRect

```kotlin
    open val baseVal: DOMRect
    open val animVal: DOMRectReadOnly
```

Exposes the JavaScript

[SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin

*An external abstract class SVGAnimatedNumberList

```kotlin
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberList
```

Exposes the JavaScript

[SVGAnimatedLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLengthList) to Kotlin

*An external abstract class SVGAnimatedLengthList

```kotlin
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthList
```

Exposes the JavaScript

[SVGStringList](https://developer.mozilla.org/en/docs/Web/API/SVGStringList) to Kotlin

```kotlin
    open val length: Int
    open val numberOfItems: Int
    fun clear(): Unit
    fun insertItemBefore(newItem: String, index: Int): String
    fun replaceItem(newItem: String, index: Int): String
    fun replaceItem(newItem: String, index: Int): String
    fun appendItem(newItem: String): String
    fun getItem(index: Int): String
```

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.get(index: Int): String? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGStringList.set(index: Int, newItem: String) { asDynamic()[index] = newItem }
```

Exposes the JavaScript

[SVGUnitTypes](https://developer.mozilla.org/en/docs/Web/API/SVGUnitTypes) to Kotlin

```kotlin
    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
    }
```

Exposes the JavaScript

[SVGTests](https://developer.mozilla.org/en/docs/Web/API/SVGTests) to Kotlin

```kotlin
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
```

Exposes the JavaScript

[SVGFitToViewBox](https://developer.mozilla.org/en/docs/Web/API/SVGFitToViewBox) to Kotlin

```kotlin
    val viewBox: SVGAnimatedRect
    val preserveAspectRatio: SVGAnimatedPreserveAspectRatio
```

Exposes the JavaScript

[SVGZoomAndPan](https://developer.mozilla.org/en/docs/Web/API/SVGZoomAndPan) to Kotlin

```kotlin
    var zoomAndPan: Short
```

```kotlin
    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
    }
```

Exposes the JavaScript

[SVGURIReference](https://developer.mozilla.org/en/docs/Web/API/SVGURIReference) to Kotlin

```kotlin
    val href: SVGAnimatedString
```

Exposes the JavaScript

[SVGSVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin

```kotlin
    val viewBox: SVGAnimatedRect
    val preserveAspectRatio: SVGAffineTransform
```

Exposes the JavaScript

[SVGText](https://developer.mozilla.org/en/docs/Web/API/SVGText) to Kotlin

```kotlin
    val requiredExtensions: SVGStringList
    val systemLanguage: SVGStringList
```

Exposes the JavaScript

[SVGTextContent](https://developer.mozilla.org/en/docs/Web/API/SVGSVGTextContent) to Kotlin

```kotlin
    val textContent: SVGTextContent
```

Exposes the JavaScript

[SVGTextContent](https://developer.mozilla.org/en/docs/Web/API/SVGSVGTextContent) to Kotlin

```kotlin
    textContent: SVGTextContent
```

Exposes the JavaScript

[SVGTextContent](https://developer.mozilla.org/en/docs/Web/API/SVGSVGTextContent) to Kotlin

```kotlin
    textContent: SVGTextContent
```

Exposes the JavaScript
external abstract class SVGSVGElement : SVGGraphicsElement, SVGFitTo viewBox, SVGZoomAndPan, 
WindowEventHandlers {
    val x: SVGAnimatedLength
    val y: SVGAnimatedLength
    val width: SVGAnimatedLength
    val height: SVGAnimatedLength
    var currentScale: Float
    var currentTranslate: DOMPoint
    fun getIntersectionList(rect: DOMRect, referenceElement: SVGElememt?): NodeList
    fun getEnclosureList(rect: DOMRect, referenceElement: SVGElememt?): NodeList
    fun checkIntersection(element: SVGElememt, rect: DOMRect): Boolean
    fun checkEnclosure(element: SVGElememt, rect: DOMRect): Boolean
    fun deselectAll()
    fun createSVGNumber(): SVGLong
    fun createSVGLength(): SVGLength
    fun createSVGAngle(): SVGAngle
    fun createSVGPoint(): DOMPoint
    fun createSVGRect(): DOMRect
    fun createSVGTransform(): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrix): SVGTransform
    fun getElementById(elementId: String): Element
    fun suspendRedraw(maxWaitMilliseconds: Int)
    fun unsuspendRedraw(suspendHandleID: Int)
    fun unsuspendRedrawAll()
    fun forceRedraw()

    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGGElement](https://developer.mozilla.org/en/docs/Web/API/SVGGElement) to Kotlin
 */
public external abstract class SVGGElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin
 */
public external abstract class SVGDefsElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin
 */
public external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val instanceRoot: SVGElement?
    open val animatedInstanceRoot: SVGElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external open class SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface SVGElementInstance {
    val correspondingElement: SVGElement?
        get() = definedExternally
    val correspondingUseElement: SVGUseElement?
        get() = definedExternally
    val ShadowAnimation(source: dynamic, newTarget: dynamic) {
        open val sourceAnimation: dynamic
    }
}

/**
 * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin
 */
public external abstract class SVGSwitchElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface GetSVGDocument {
    fun getSVGDocument(): Document
}

/**
 * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
 */
public external abstract class SVGStyleElement : SVGElement, LinkStyle {
    open var type: String
    open var media: String
    open var title: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
PUBLIC SOURCE USED IN DNAC PLATFORM

// Exposes the JavaScript [SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin
public external abstract class SVGTransform {
    open val type: Short
    open val matrix: DOMMatrix
    open val angle: Float
    fun setMatrix(matrix: DOMMatrixReadOnly)
    fun setTranslate(tx: Float, ty: Float)
    fun setScale(sx: Float, sy: Float)
    fun setRotate(angle: Float, cx: Float, cy: Float)
    fun setSkewX(angle: Float)
    fun setSkewY(angle: Float)

    companion object {
        val SVG_TRANSFORM_UNKNOWN: Short
        val SVG_TRANSFORM_MATRIX: Short
        val SVG_TRANSFORM_TRANSLATE: Short
        val SVG_TRANSFORM_SCALE: Short
        val SVG_TRANSFORM_ROTATE: Short
        val SVG_TRANSFORM_SKEWX: Short
        val SVG_TRANSFORM_SKEWY: Short
    }
}

/**
 * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
 */
public external abstract class SVGTransformList {
    open val length: Int
    open val numberOfItems: Int

    fun clear()
    fun initialize(newItem: SVGTransform): SVGTransform
    fun insertItemBefore(newItem: SVGTransform, index: Int): SVGTransform
    fun replaceItem(newItem: SVGTransform, index: Int): SVGTransform
    fun removeItem(index: Int): SVGTransform
    fun appendItem(newItem: SVGTransform): SVGTransform
    fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
    fun consolidate(): SVGTransform?
    fun getItem(index: Int): SVGTransform

    @kotlin.internal.InlineOnly
    public inline operator fun SVGTransformList.get(index: Int): SVGTransform? = asDynamic()[index]

    @kotlin.internal.InlineOnly
    public inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform) { asDynamic()[index] = newItem }

    /**
     * Exposes the JavaScript [SVGPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGPathElement) to Kotlin
     */
    public external abstract class SVGPathElement : SVGGeometryElement {
        companion object {
            val
Open Source Used In DNAC Platform Arica 13442

`ELEMENT_NODE`: Short  
`ATTRIBUTE_NODE`: Short  
`TEXT_NODE`: Short  
`CDATA_SECTION_NODE`: Short  
`ENTITY_REFERENCE_NODE`: Short  
`ENTITY_NODE`: Short  
`PROCESSING_INSTRUCTION_NODE`: Short  
`COMMENT_NODE`: Short  
`DOCUMENT_NODE`: Short  
`DOCUMENT_TYPE_NODE`: Short  
`DOCUMENT_FRAGMENT_NODE`: Short  
`NOTATION_NODE`: Short  
`DOCUMENT_POSITION_DISCONNECTED`: Short  
`DOCUMENT_POSITION_PRECEDING`: Short  
`DOCUMENT_POSITION_FOLLOWING`: Short  
`DOCUMENT_POSITION_CONTAINS`: Short  
`DOCUMENT_POSITION_CONTAINED_BY`: Short  

`DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC`: Short  

/**
 * Exposes the JavaScript [SVGRectElement](https://developer.mozilla.org/en/docs/Web/API/SVGRectElement) to Kotlin
 */
public external abstract class SVGRectElement : SVGGeometryElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val rx: SVGAnimatedLength
    open val ry: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short  
        val ATTRIBUTE_NODE: Short  
        val TEXT_NODE: Short  
        val CDATA_SECTION_NODE: Short  
        val ENTITY_REFERENCE_NODE: Short  
        val ENTITY_NODE: Short  
        val PROCESSING_INSTRUCTION_NODE: Short  
        val COMMENT_NODE: Short  
        val DOCUMENT_NODE: Short  
        val DOCUMENT_TYPE_NODE: Short  
        val DOCUMENT_FRAGMENT_NODE: Short  
        val NOTATION_NODE: Short  
        val DOCUMENT_POSITION_DISCONNECTED: Short  
        val DOCUMENT_POSITION_PRECEDING: Short  
        val DOCUMENT_POSITION_FOLLOWING: Short  
        val DOCUMENT_POSITION_CONTAINS: Short  
        val DOCUMENT_POSITION_CONTAINED_BY: Short  
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short  
    }
}

/**
 * Exposes the JavaScript [SVGCircleElement](https://developer.mozilla.org/en/docs/Web/API/SVGCircleElement) to Kotlin
 */
public external abstract class SVGCircleElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short  
        val ATTRIBUTE_NODE: Short  
        val TEXT_NODE: Short  
        val CDATA_SECTION_NODE: Short  
        val ENTITY_REFERENCE_NODE: Short  
        val ENTITY_NODE: Short  
        val PROCESSING_INSTRUCTION_NODE: Short  
        val COMMENT_NODE: Short  
        val DOCUMENT_NODE: Short  
        val DOCUMENT_TYPE_NODE: Short  
        val DOCUMENT_FRAGMENT_NODE: Short  
        val NOTATION_NODE: Short  
        val DOCUMENT_POSITION_DISCONNECTED: Short  
        val DOCUMENT_POSITION_PRECEDING: Short  
        val DOCUMENT_POSITION_FOLLOWING: Short  
        val DOCUMENT_POSITION_CONTAINS: Short  
        val DOCUMENT_POSITION_CONTAINED_BY: Short  
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short  
    }
}

/**
 * Exposes the JavaScript [SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin
 */
public external abstract class SVGEllipseElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val rx: SVGAnimatedLength
    open val ry: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short  
        val ATTRIBUTE_NODE: Short  
        val TEXT_NODE: Short  
        val CDATA_SECTION_NODE: Short  
        val ENTITY_REFERENCE_NODE: Short  
        val ENTITY_NODE: Short  
        val PROCESSING_INSTRUCTION_NODE: Short  
        val COMMENT_NODE: Short  
        val DOCUMENT_NODE: Short  
        val DOCUMENT_TYPE_NODE: Short  
        val DOCUMENT_FRAGMENT_NODE: Short  
        val NOTATION_NODE: Short  
        val DOCUMENT_POSITION_DISCONNECTED: Short  
        val DOCUMENT_POSITION_PRECEDING: Short  
        val DOCUMENT_POSITION_FOLLOWING: Short  
        val DOCUMENT_POSITION_CONTAINS: Short  
        val DOCUMENT_POSITION_CONTAINED_BY: Short  
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short  
    }
}

/**
 * Exposes the JavaScript [SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin
 */
public external abstract class SVGLineElement : SVGGeometryElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength
}
val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE:
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin */
public external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin */
public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
    fun removeItem(index: Int): DOMPoint
    fun appendItem(newItem: DOMPoint): DOMPoint
    fun getItem(index: Int): DOMPoint
}

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin */
public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin */
public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```kotlin
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
```
Open Source Used In DNAC Platform Arica 13446

[SVGForeignObjectElement](https://developer.mozilla.org/en/docs/Web/API/SVGFinePath) to Kotlin

*public external abstract class SVGFinePath : SVGGraphicsElement {

val x: SVGAnimatedLength
val y: SVGAnimatedLength
val width: SVGAnimatedLength
val height: SVGAnimatedLength
val companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript

[SVGSolidcolorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidcolorElement) to Kotlin

*public external abstract class SVGSolidcolorElement : SVGElement {

val companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript

[SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin

*public external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {

val gradientUnits: SVGAnimatedEnumeration
val gradientTransform: SVGAnimatedTransformList
val spreadMethod: SVGAnimatedEnumeration
val companion object {
    val SVG_SPREADMETHOD_UNKNOWN: Short
    val SVG_SPREADMETHOD_PAD: Short
    val SVG_SPREADMETHOD_REFLECT: Short
    val SVG_SPREADMETHOD_REPEAT: Short
    val SVG_UNIT_TYPE_UNKNOWN: Short
    val SVG_UNIT_TYPE_USERSPACEONUSE: Short
    val
**`SVGUNITTYPEOBJECTBOUNDINGBOX`: Short
`val ELEMENT_NODE`: Short
`val ATTRIBUTE_NODE`: Short
`val TEXT_NODE`: Short
`val CDATA_SECTION_NODE`: Short
`val ENTITY_REFERENCE_NODE`: Short
`val ENTITY_NODE`: Short

**`PROCESSING_INSTRUCTION_NODE`: Short
`val COMMENT_NODE`: Short
**`DOCUMENT_NODE`: Short
`val DOCUMENT_TYPE_NODE`: Short

**`DOCUMENT_FRAGMENT_NODE`: Short
`val NOTATION_NODE`: Short

**`DOCUMENTPOSITION_DISCONNECTED`: Short
`val DOCUMENTPOSITION_PRECEDING`: Short
`val DOCUMENTPOSITION_FOLLOWING`: Short
`val DOCUMENTPOSITION_CONTAINS`: Short
`val DOCUMENTPOSITIONCONTAINEDBY`: Short

**`DOCUMENTPOSITIONIMPLEMENTATIONSPECIFIC`: Short

`@` Exposes the JavaScript [SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to Kotlin

```kotlin
public external abstract class SVGLinearGradientElement : SVGGradientElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val SVGSPREADMETHODUNKNOWN: Short
        val SVGSPREADMETHODPAD: Short
        val SVGSPREADMETHODREFLECT: Short
        val SVGSPREADMETHODREPEAT: Short
        val SVGUNITTYPEUNKNOWN: Short
        val SVGUNITTYPEUSERSPACEONUSE: Short
        val SVGUNITTYPEOBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENTPOSITION_DISCONNECTED: Short
        val DOCUMENTPOSITION_PRECEDING: Short
        val DOCUMENTPOSITION_FOLLOWING: Short
        val DOCUMENTPOSITION_CONTAINS: Short
        val DOCUMENTPOSITIONCONTAINEDBY: Short
        val DOCUMENTPOSITIONIMPLEMENTATIONSPECIFIC: Short
    }
}
```

**`SVGRadialGradientElement`: Short

```kotlin
public external abstract class SVGRadialGradientElement : SVGGradientElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength
    open val fx: SVGAnimatedLength
    open val fy: SVGAnimatedLength
    open val fr: SVGAnimatedLength

    companion object {
        val SVGSPREADMETHODUNKNOWN: Short
        val SVGSPREADMETHODPAD: Short
        val SVGSPREADMETHODREFLECT: Short
        val SVGSPREADMETHODREPEAT: Short
        val SVGUNITTYPEUNKNOWN: Short
        val SVGUNITTYPEUSERSPACEONUSE: Short
        val SVGUNITTYPEOBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENTPOSITION_DISCONNECTED: Short
        val DOCUMENTPOSITION_PRECEDING: Short
        val DOCUMENTPOSITION_FOLLOWING: Short
        val DOCUMENTPOSITION_CONTAINS: Short
        val DOCUMENTPOSITIONCONTAINEDBY: Short
        val DOCUMENTPOSITIONIMPLEMENTATIONSPECIFIC: Short
    }
}
```

**`SVGMeshGradientElement`: Short

```kotlin
public external abstract class SVGMeshGradientElement : SVGGradientElement {

    companion object {
        val SVGSPREADMETHODUNKNOWN: Short
        val SVGSPREADMETHODPAD: Short
        val SVGSPREADMETHODREFLECT: Short
        val SVGSPREADMETHODREPEAT: Short
        val SVGUNITTYPEUNKNOWN: Short
        val SVGUNITTYPEUSERSPACEONUSE: Short
        val SVGUNITTYPEOBJECTBOUNDINGBOX: Short
    }
}
```
ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external abstract class SVGMeshrowElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGMeshpatchElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGStopElement](https://developer.mozilla.org/en/docs/Web/API/SVGStopElement) to Kotlin
 */
public external abstract class SVGStopElement : SVGElement {
    open val offset: SVGAnimatedNumber

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGPatternElement](https://developer.mozilla.org/en/docs/Web/API/SVGPatternElement) to Kotlin
 */
public external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {
    open val patternUnits: SVGAnimatedEnumeration
    open val patternContentUnits: SVGAnimatedEnumeration
    open val patternTransform: SVGAnimatedTransformList
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

public external abstract class SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGCursorElement](https://developer.mozilla.org/en/docs/Web/API/SVGCursorElement) to Kotlin
 */
public external abstract class SVGCursorElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin
 */
public external abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLOrSVGScriptElement {
    open var type: String
    open var crossOrigin: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [SVGElement](https://developer.mozilla.org/en/docs/Web/API/SVGElement) to Kotlin */
public external abstract class SVGElement : SVGAElement, SVGURIReference {
    open val target: SVGAnimatedString
    open val download: SVGAnimatedString
    open val rel: SVGAnimatedString
    open val relList: SVGAnimatedString
    open val hreflang: SVGAnimatedString
    open val type: SVGAnimatedString

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGViewElement](https://developer.mozilla.org/en/docs/Web/API/SVGViewElement) to Kotlin */
public external abstract class SVGViewElement : SVGElement, SVGFitToViewBox, SVGZoomAndPan {
    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
definedExternally
set(value) = definedExternally
var silent: Boolean? /* = false */
get() =
definedExternally
set(value) = definedExternally
var noscreen: Boolean? /* = false */
get() =
definedExternally
set(value) = definedExternally
var requireInteraction: Boolean? /* = false */
get() =
definedExternally
set(value) = definedExternally
var sticky: Boolean? /* = false */
get() =
definedExternally
set(value) = definedExternally
var data: Any? /* = null */
get() =
definedExternally
set(value) = definedExternally
var actions: Array<NotificationAction>? /* = arrayOf() */
get() =
definedExternally
set(value) = definedExternally
var icon: String?
get() =
definedExternally
set(value) = definedExternally
var image: String?
get() =
definedExternally
set(value) = definedExternally
var badge: String?
get() =
definedExternally
set(value) = definedExternally
var sound: String?
get() =
definedExternally
set(value) = definedExternally
var title: String?
get() =
definedExternally
set(value) = definedExternally
var tag: String?
get() =
definedExternally
set(value) = definedExternally
var reload: Boolean? /* = false */
get() =
definedExternally
set(value) = definedExternally
var delay: Number?
get() =
definedExternally
set(value) = definedExternally
var data: Any?
get() =
definedExternally
set(value) = definedExternally
var actions: Array<NotificationAction>? /* = arrayOf() */
get() =
NotificationDirection.Companion.LTR: NotificationDirection get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()

NotificationDirection.Companion.RTL: NotificationDirection get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.LTR: NotificationDirection get() = "ltr".asDynamic().unsafeCast<NotificationDirection>()

public inline val NotificationDirection.Companion.RTL: NotificationDirection get() = "rtl".asDynamic().unsafeCast<NotificationDirection>()

package org.w3c.workers

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin
 */
public external abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open val scriptURL: String
    open val state: ServiceWorkerState
    open var onstatechange: ((Event) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript [ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to Kotlin
 */
public external abstract class ServiceWorkerRegistration : EventTarget {
    open val installing: ServiceWorker?
    open val waiting: ServiceWorker?
    open val active: ServiceWorker?
    open val scope: String
    open var onupdatefound: ((Event) -> dynamic)?
    open val APISpace: dynamic
    fun update(): Promise<Unit>
    fun unregister(): Promise<Boolean>
    fun showNotification(title: String, options: NotificationOptions = definedExternally): Promise<Unit>
    fun getNotifications(filter: GetNotificationOptions = definedExternally): Promise<Array<Notification>>
    fun methodName(): Promise<dynamic>
}

/**
 * Exposes the JavaScript [ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin
 */
public external abstract class ServiceWorkerContainer : EventTarget {
    open val controller: ServiceWorker?
    open val ready: Promise<ServiceWorkerRegistration>
    open var oncontrollerchange: ((Event) -> dynamic)?
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun register(scriptURL: String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>
    fun getRegistration(clientURL: String = definedExternally): Promise<Any?>
    fun getRegistrations(): Promise<Array<ServiceWorkerRegistration>>
    fun startMessages()
}

public external interface RegistrationOptions {
    var scope: String?
        get() = definedExternally
        set(value) = definedExternally
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {
    val o = js("({})")
    o["scope"] = scope
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript [ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to Kotlin
 */
public external open class ServiceWorkerMessageEvent(type: String, eventInitDict: ServiceWorkerMessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val source: UnionMessagePortOrServiceWorker?
    open val ports: Array<MessagePort>?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ServiceWorkerMessageEventInit : EventInit {
    var data: Any?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>?
        get() = definedExternally
        set(value) = definedExternally
}

Open Source Used In DNAC Platform Arica 13453
Open Source Used In DNAC Platform Arica 13454

```kotlin
@kotlin.internal.InlineOnly
    val o = js("{}")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
}

/**
 * Exposes the JavaScript
 */
public external class ServiceWorkerGlobalScope {
    fun openGlobalScope(): Promise<ServiceWorkerGlobalScope>
Open Source Used In DNAC Platform Arica 13455

[FetchEvent](https://developer.mozilla.org/en/docs/Web/API/FetchEvent) to Kotlin

```
public external open class FetchEvent(type: String, eventInitDict: FetchEventInit) : ExtendableEvent {
    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["request"] = request
        o["clientId"] = clientId
        o["isReload"] = isReload
        o["bubbles"] = bubbles
        o["Cancelable"] = cancelable
        return o
    }
```

```kotlin
/**
 * Exposes the JavaScript [ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin
 */
public external open class ExtendableMessageEvent(type: String, eventInitDict: ExtendableMessageEventInit) : ExtendableEvent {
    @kotlin.internal.InlineOnly
    inline fun ExtendableMessageEventInit() {
        val o = js("({})")
        return o
    }
```

```
/**
 * Exposes the JavaScript [Cache](https://developer.mozilla.org/en/docs/Web/API/Cache) to Kotlin
 */
public external abstract class Cache
```
fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):
Promise<Array<Response>>
fun add(request: dynamic): Promise<Unit>
fun addAll(requests: Array<dynamic>): Promise<Unit>
fun put(request: dynamic, response: Response): Promise<Unit>
fun delete(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>
fun keys(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):
Promise<Array<Request>>

fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):
Promise<Array<Response>>
fun add(request: dynamic): Promise<Unit>
fun addAll(requests: Array<dynamic>): Promise<Unit>
fun put(request: dynamic, response: Response): Promise<Unit>
fun delete(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>
fun keys(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally):
Promise<Array<Request>>

public external interface CacheQueryOptions {
    var ignoreSearch: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ignoreMethod: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ignoreVary: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cacheName: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ignoreSearch"] = ignoreSearch
    o["ignoreMethod"] = ignoreMethod
    o["ignoreVary"] = ignoreVary
    o["cacheName"] = cacheName
    return o
}

public external interface CacheBatchOperation {
    var type: String?
        get() = definedExternally
        set(value) = definedExternally
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var response: Response?
        get() = definedExternally
        set(value) = definedExternally
    var options: CacheQueryOptions?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["type"] = type
    o["request"] = request
    o["response"] = response
    o["options"] = options
    return o
}

/**
 * Exposes the JavaScript
[CacheStorage](https://developer.mozilla.org/en/docs/Web/API/CacheStorage) to Kotlin
 */
public external abstract class CacheStorage {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun has(cacheName: String): Promise<Boolean>
    fun open(cacheName: String):
Promise<Cache>
    fun delete(cacheName: String): Promise<Boolean>
    fun keys():
Promise<Array<String>>
}

public external open class FunctionalEvent : ExtendableEvent {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface UnionMessagePortOrServiceWorker

public external interface UnionClientOrMessagePortOrServiceWorker

/* please, don't implement this interface!*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ServiceWorkerState {
    companion object
}

public inline val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() = "installing".asDynamic().unsafeCast<ServiceWorkerState>()
public inline val ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() = "installed".asDynamic().unsafeCast<ServiceWorkerState>()
public inline val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() = "activating".asDynamic().unsafeCast<ServiceWorkerState>()
public inline val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() = "activated".asDynamic().unsafeCast<ServiceWorkerState>()
public inline val ServiceWorkerState.Companion.REDUndant: ServiceWorkerState get() = "redundant".asDynamic().unsafeCast<ServiceWorkerState>()

/* please, don't implement this interface!*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface FrameType {
    companion object
}

public inline val FrameType.Companion.AUXILIARY: FrameType get() = "auxiliary".asDynamic().unsafeCast<FrameType>()
public inline val FrameType.Companion.TOP_LEVEL: FrameType get() = "top-level".asDynamic().unsafeCast<FrameType>()
public inline val FrameType.Companion.NESTED: FrameType get() = "nested".asDynamic().unsafeCast<FrameType>()
public inline val FrameType.Companion.NONE: FrameType get() = "none".asDynamic().unsafeCast<FrameType>()

public external interface ClientType {
    companion object
}

public inline val ClientType.Companion.WINDOW: ClientType get() = "window".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.WORKER: ClientType get() = "worker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.SHAREDWORKER: ClientType get() = "sharedworker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.ALL: ClientType get() = "all".asDynamic().unsafeCast<ClientType>()

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details
package org.w3c.xhr

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*

/**
 * Exposes the JavaScript XMLHttpRequestEventTarget to Kotlin
 */
public external abstract class XMLHttpRequestEventTarget : EventTarget {
    open var onloadstart: ((ProgressEvent) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onabort: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onload: ((Event) -> dynamic)?
    open var ontimeout: ((Event) -> dynamic)?
    open var onloadend: ((Event) -> dynamic)?
}

public external abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget

/**
 * Exposes the JavaScript XMLHttpRequest to Kotlin
 */
public external open class XMLHttpRequest : XMLHttpRequestEventTarget {
    var onreadystatechange: ((Event) -> dynamic)?
    open val readyState: Short
    var timeout: Int
    var withCredentials: Boolean
    open val upload: XMLHttpRequestUpload
    open val responseURL: String
    open val status: Short
    open val statusText: String
    var responseType: XMLHttpRequestResponseType
    open val response: Any?
    open val responseText: String
    open val responseXML: Document?

    fun open(method: String, url: String)
    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String?
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Short
        val OPENED: Short
        val HEADERS_RECEIVED: Short
        val LOADING: Short
        val DONE: Short
    }
}

/**
 * Exposes the JavaScript FormData to Kotlin
 */
public external open class FormData(form: HTMLFormElement = definedExternally) {
    fun append(name: String, value: String)
    fun append(name: String, value: Blob, filename: String = definedExternally)
    fun delete(name: String)
    fun get(name: String): dynamic
    fun getAll(name: String): Array<dynamic>
    fun has(name: String): Boolean
    fun set(name: String, value: String)
    fun set(name: String, value: Blob, filename: String = definedExternally)

    companion object {
        val UNSENT: Any
        val OPENED: Any
        val HEADERS_RECEIVED: Any
        val LOADING: Any
        val DONE: Any
    }
}

/**
 * Exposes the JavaScript ProgressEvent to Kotlin
 */
public external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) : Event {
    open val lengthComputable: Boolean
    open val loaded: Number
    open val total: Number

    companion object {
        val NONE: ProgressEventInit
        val CAPTURING_PHASE: ProgressEventInit
        val AT_TARGET: ProgressEventInit
        val BUBBLING_PHASE: ProgressEventInit
    }

    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Any
        val OPENED: Any
        val HEADERS_RECEIVED: Any
        val LOADING: Any
        val DONE: Any
    }
}
definedExternally

    val o = js("{}") asDynamic().unsafeCast<XMLHttpRequestResponseType>()
    o["lengthComputable"] = lengthComputable
    o["loaded"] = loaded
    o["total"] = total
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don’t implement this interface! */
@kotlin.internal.InlineOnly
    val o = js("{}")
    o["lengthComputable"] = lengthComputable
    o["loaded"] = loaded
    o["total"] = total
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don’t implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface XMLHttpRequestResponseType {
    companion object
}

public inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() = "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() = "arraybuffer".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = "blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = "document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = "json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = "text".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

package kotlin

import kotlin.annotation.AnnotationRetention.BINARY
import kotlin.annotation.AnnotationRetention.SOURCE
import kotlin.reflect.KClass

/**
 * Signals that the annotated
 * annotation class is a marker of an experimental API.
 *
 * Any declaration annotated with that marker is
 * considered an experimental declaration and its call sites should accept the experimental aspect of it either by
 * using [UseExperimental], or by being annotated with that marker themselves, effectively causing further
 * propagation of that experimental aspect.
 *
 * This class is deprecated in favor of a more general approach
 * provided by [RequiresOptIn]/[OptIn].
 */
@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.2")
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
@Deprecated("Please use RequiresOptIn instead.", ReplaceWith("OptIn(*markerClass)", "kotlin.OptIn"))
public annotation class Experimental(val level: Level = Level.ERROR) {
    /**
     * Severity of the
     * diagnostic that should be reported on usages of experimental API which did not explicitly accept the experimental
     * aspect of that API either by using [UseExperimental] or by being annotated with the corresponding marker
     * annotation.
     */
    public enum class Level {
        /** Specifies that a warning should be reported on incorrect
         * usages of this experimental API. */
        WARNING,
        /** Specifies that an error should be reported on incorrect
         * usages of this experimental API. */
        ERROR
    }

    /**
     * Allows to use experimental API
denoted by the given markers in the annotated file, declaration, or expression.
     * If a declaration is annotated with [UseExperimental], its usages are **not** required to opt-in to that experimental API.
     * * This class is deprecated in favor of a more general approach provided by [RequiresOptIn]/[OptIn].
     */
    @Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE, TYPEALIAS)
    @Retention(SOURCE)
    @SinceKotlin("1.2")
    @RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPILE

    ReplaceWith("OptIn(*markerClass)", "kotlin.OptIn")
}

public annotation class UseExperimental(
    vararg val markerClass: KClass<out Annotation>
)

internal annotation class WasExperimental(
    vararg val markerClass: KClass<out Annotation>
)
annotation marks the standard library API that is considered experimental and is not subject to the general compatibility guarantees given for the standard library. Any usage of a declaration annotated with `@ExperimentalStdlibApi` must be accepted either by annotating that usage with the `[OptIn] annotation, e.g. `@OptIn(ExperimentalStdlibApi::class)`, or by using the compiler argument `-Xopt-in=kotlin.ExperimentalStdlibApi`. Any *experimental status from itself and advance SinceKotlin to 1.4*

package kotlin

import kotlin.annotation.AnnotationTarget.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Allows to infer generic type arguments of a function from the calls in the annotated function parameter of that function. *When this annotation is placed on a generic function parameter of a function, it enables to infer the type arguments of that function from the lambda body passed to that parameter. * The calls that affect inference are either members of the receiver type of an annotated function parameter or annotations for that type. * Extensions for that type. The extensions must be themselves annotated with `@BuilderInference`. * Example: we declare a function that takes a type `T` as an argument and yields a sequence of `T`:
 * ```
 * fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T>
 * ```
 * and use it like:
 * ```
 * val result = sequence {
 * yield("result")
 * ```
 * Here the type argument of the resulting sequence is inferred to `String` from the argument of the lambda function, which is inside the lambda passed to `sequence`. * Note: this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.

@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
@ExperimentalTypeInference
public annotation class BuilderInference

@Target(CLASS, ANNOTATION_CLASS, PROPERTY, FIELD, LOCAL_VARIABLE, VALUE_PARAMETER, PROPERTY_GETTER, PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
@RequiresOptIn
public annotation class ExperimentalMultiplatform

/**
 * Marks an expected annotation class that it isn't required to have actual counterparts in all platforms. * This annotation is only applicable to `expect` annotation classes in multi-platform projects and marks that class as "optional". * This annotation allows to use it in Kotlin's standard library only if there are no actual counterparts in all platforms. * Optional expected class is allowed to have no corresponding actual class on the platform. * Optional annotations can only be used to annotate something, not as types in signatures. If an optional annotation has no corresponding actual class on a platform, the annotation entries where it's used are simply erased when
compiling code on that platform.

*Note: this annotation is experimental, see [ExperimentalMultiplatform] on how to opt-in for it.*

```kotlin
@Target(ANNOTATION_CLASS)
@Retention(RetentionPolicy.BINARY)
@ExperimentalMultiplatform
@RequireKotlin("1.2.50", versionKind = RequireKotlinVersionKind.COMPLIER_VERSION)
@Public
@OptionalExpectation
```

The default message is: 

```
This declaration is experimental and its usage should be marked with 'Marker
or '@OptIn(Marker::class)"', where 'Marker' is the opt-in requirement marker.
```

The default message is: 

```
This declaration is experimental and its usage should be marked with 'Marker
or '@OptIn(Marker::class)"', where 'Marker' is the opt-in requirement marker.
```

```kotlin
package kotlin

import kotlin.annotation.AnnotationRetention.BINARY
import kotlin.annotation.AnnotationRetention.SOURCE
import kotlin.internal.RequireKotlin
import kotlin.internal.RequireKotlinVersionKind
import kotlin.reflect.KClass

/**
 * Signals that the annotated
 * annotation class is a marker of an API that requires an explicit opt-in.
 * * Call sites of any declaration annotated
 * with that marker should opt in to the API either by using [OptIn], * or by being annotated
 * with that marker themselves, effectively creating further propagation of the opt-in requirement.
 * * This class requires opt-in itself
 * and can only be used with the compiler argument `-Xopt-in=kotlin.RequiresOptIn
 * @property message message to be reported on usages of API without an explicit opt-in, or empty string for the default message.
 * * The default message is: 
 * `This declaration is experimental and its usage should be marked with 'Marker'
 * or '@OptIn(Marker::class)"', where 'Marker' is the opt-in requirement marker.
 * @property level specifies
 * how usages of API without an explicit opt-in are reported in code.
 */
@Target(ANNOTATION_CLASS)
@Retention(BINARY)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind = RequireKotlinVersionKind.COMPILER_VERSION)
public annotation class RequiresOptIn(
    val message: String = "",
    val level: Level = Level.ERROR
) {
    /**
     * Severity of the diagnostic that
     * should be reported on usages which did not explicitly opted into
     * the API either by using [OptIn] or by being
     * annotated with the corresponding marker annotation.
     * * This class requires opt-in itself
     * and can only be used with the compiler argument `-Xopt-in=kotlin.RequiresOptIn
     * @property message message to be reported on usages of API without an explicit opt-in, or empty string for the default message.
     * * The default message is: 
     * `This declaration is experimental and its usage should be marked with 'Marker'
     * or '@OptIn(Marker::class)"', where 'Marker' is the opt-in requirement marker.
     * @property level specifies
     * how usages of API without an explicit opt-in are reported in code.
 */
@Target(ANNOTATION_CLASS)
@Retention(RetentionPolicy.BINARY)
@SinceKotlin("1.3")
@RequireKotlin("1.3.70", versionKind = RequireKotlinVersionKind.COMPLIER_VERSION)
public annotation class RequiresOptIn

/**
 * Provides a skeletal
 * implementation of the read-only 
 * interface.
 * @param E the type of elements contained in the
 * collection. The collection is covariant on its element type.
 */
@SinceKotlin("1.1")
public abstract class AbstractCollection<out E> protected constructor() : Collection<E> {
    abstract override val size: Int
    abstract
    override fun iterator(): Iterator<E>

    override fun contains(element: @UnsafeVariance E): Boolean = any { it == element }

    override fun containsAll(elements: Collection<@UnsafeVariance E>): Boolean = elements.all { it == element }

    override fun isEmpty(): Boolean = size == 0

    override fun toString(): String = joinToString(", ", "[", "]") {
        if (it === this) "(this Collection)" else it.toString()
    }

    /**
     * Returns new array of type `Array<Any?>` with
     * the elements of this collection.
     */
    @JsName("toArray")
    protected open fun toArray(): Array<Any?> = copyToArrayImpl(this)

    /**
     * Fills the provided 
     * array or creates new array of the same type
     * and
     * fills it with the elements of this collection.
     */
    protected open fun copyToArrayImpl(array: Array<Any?>): Array<Any?>
```

```kotlin
open class AbstractCollection<E>(
    @UnsafeVariance private val element: E
) : Collection<E>() {
    override val size: Int
    override fun iterator(): Iterator<E>

    override fun contains(element: E): Boolean = any { it == element }

    override fun containsAll(elements: Collection<E>): Boolean = elements.all { it == element }

    override fun isEmpty(): Boolean = size == 0

    override fun toString(): String = joinToString(", ", "[", "]") {
        if (it === this) "(this Collection)" else it.toString()
    }

    /**
     * Returns new array of type `Array<Any?>` with
     * the elements of this collection.
     */
    protected open fun copyToArrayImpl(array: Array<Any?>): Array<Any?>
```

```kotlin
open class AbstractCollection<E>(
    @UnsafeVariance private val element: E
) : Collection<E>() {
    override val size: Int
    override fun iterator(): Iterator<E>

    override fun contains(element: E): Boolean = any { it == element }

    override fun containsAll(elements: Collection<E>): Boolean = elements.all { it == element }

    override fun isEmpty(): Boolean = size == 0

    override fun toString(): String = joinToString(", ", "[", "]") {
        if (it === this) "(this Collection)" else it.toString()
    }

    /**
     * Returns new array of type `Array<Any?>` with
     * the elements of this collection.
     */
    protected open fun copyToArrayImpl(array: Array<Any?>): Array<Any?>
```

```kotlin
open class AbstractCollection<E>(
    @UnsafeVariance private val element: E
) : Collection<E>() {
    override val size: Int
    override fun iterator(): Iterator<E>

    override fun contains(element: E): Boolean = any { it == element }

    override fun containsAll(elements: Collection<E>): Boolean = elements.all { it == element }

    override fun isEmpty(): Boolean = size == 0

    override fun toString(): String = joinToString(", ", "[", "]") {
        if (it === this) "(this Collection)" else it.toString()
    }

    /**
     * Returns new array of type `Array<Any?>` with
     * the elements of this collection.
     */
    protected open fun copyToArrayImpl(array: Array<Any?>): Array<Any?>
```
AbstractCollection<V>() { 
    override operator fun contains(element: @UnsafeVariance V): Boolean =
        containsValue(element)

    override operator fun iterator(): Iterator<V> {
        return object : Iterator<V> {
            override fun hasNext(): Boolean = false
            override fun next(): V = entryIterator.next().value
        }
    }

    override val size: Int get() = this@AbstractMap.size
}

private fun implFindEntry(key: K): Map.Entry<K, V>? = entries.firstOrNull { it.key == key }

internal companion object {

    internal fun entryHashCode(e: Map.Entry<*, *>): Int = with(e) { (key?.hashCode() ?: 0) xor (value?.hashCode() ?: 0) }

    internal fun entryToString(e: Map.Entry<*, *>): String = with(e) {
        "$key=$value"
    }

    internal fun entryEquals(e: Map.Entry<*, *>, other: Any?): Boolean {
        if (other !is Map.Entry<*, *>) return false
        return e.key == other.key && e.value == other.value
    }

}
Open Source Used In DNAC Platform Arica

```

// Also the circular array can store at least one more element.

// Elements in circular array,
// and tail is the index following the last element.

// At this point the
#
//   2. `head` is greater than `tail`:    
//   
// Elements in circular array lay in 2 ways:
//   1. `head` is less than `tail`:       

// At this point the
#
//   2. `head` is greater than `tail`:    
//     ...
```
val element = elementData[index]

if (predicate(element as E))
    val elementData[newTail + 1] = element
    newTail = positiveMod(newTail + 1, elementData.size)  
else 
    val element = elementData[index]
    elementData[index] = null

@Suppress("UNCHECKED_CAST")  
for (index in head until tail) 
    {  
        val element = elementData[index]
        elementData[index] = null

        @Suppress("UNCHECKED_CAST")
        if (predicate(element as E))
            elementData[newTail] = element
            newTail = incremented(newTail)  
        else
            return modified
    }

    @Suppress("UNCHECKED_CAST")
    if (predicate(element as E))
        elementData[newTail] = element
        newTail = incremented(newTail)
    else
        return modified

    for (index in head until tail) 
        {  
            val element = elementData[index]
            elementData[index] = null

            @Suppress("UNCHECKED_CAST")
            if (predicate(element as E))
                elementData[newTail] = element
                newTail = incremented(newTail)
            else
                return modified
        }

    return modified

    public override fun clear() {
        val tail = internalIndex(size)
        if (head < tail) {
            elementData.fill(null, head, tail)
        } else if (isNotEmpty()) {
            elementData.fill(null, head, elementData.size)
            elementData.fill(null, 0, tail)
        }
        head = 0
        size = 0
        return modified
    }

    // For testing only
    private fun internalStructure(structure: (head: Int, elements: Array<Any?>) -> Unit) {
        val tail = internalIndex(size)

        if (isEmpty()) {
            structure(head, emptyArray())
            return
        }

        val elements = arrayOfNulls<Any?>(size)
        if (head < tail) {
            elementData.copyInto(elements, startIndex = head, endIndex = tail)
            structure(head, elements)
        } else {
            elementData.copyInto(elements, startIndex = head)
            elementData.copyInto(elements, elementData.size - head, startIndex = 0, endIndex = tail)
            structure(head - elementData.size, elements)
        }
    }

    public fun <T> Array<out Array<out T>>.flatten(): List<T> {
        val result = ArrayList<T>(sumBy { it.size })
        for (element in this) {
            result.addAll(element)
        }
        return result
    }

    public fun <T, R> Array<out Pair<T, R>>.unzip(): Pair<List<T>, List<R>> {
        val listT = ArrayList<T>(size)
        val listR = ArrayList<R>(size)
        for (pair in this) {
            listT.add(pair.first)
            listR.add(pair.second)
        }
        return listT to listR
    }

    public fun <C : Array<*>, C : R> C.ifEmpty(defaultValue: () -> R): R
        where C : Array<*>, C : R =
        if (isEmpty())
            defaultValue() else
            this

    public fun <T> Array<out T>.contentDeepEqualsImpl(other: Array<out T>): Boolean {
        if (this === other) return true
        if (this.size != other.size) return false

        for (i in indices) {
            val v1 = this[i]
            val v2 = other[i]

            if (v1 == null || v2 == null) {
                return false
            }

            when {
                v1 is Array<*>
                && v2 is Array<*> -> if (!v1.contentDeepEquals(v2)) return false
                v1 is ByteArray
                && v2 is ByteArray -> if (!v1.contentEquals(v2)) return false
                v1 is ShortArray
                && v2 is ShortArray -> if (!v1.contentEquals(v2)) return false
                v1 is IntArray
                && v2 is IntArray -> if (!v1.contentEquals(v2)) return false
                v1 is LongArray
                && v2 is LongArray -> if (!v1.contentEquals(v2)) return false
            }
        }
        return true
    }
}
return false
v1 is FloatArray && v2 is FloatArray -> if (!v1.contentEquals(v2)) return false
v1 is DoubleArray && v2 is DoubleArray -> if (!v1.contentEquals(v2)) return false
v1 is CharArray -> if (!v1.contentEquals(v2)) return false
v1 is BooleanArray && v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false
v1 is UByteArray && v2 is UByteArray -> if (!v1.contentEquals(v2)) return false
v1 is UShortArray && v2 is UShortArray -> if (!v1.contentEquals(v2)) return false
v1 is.UIntArray && v2 is.UIntArray -> if (!v1.contentEquals(v2)) return false
v1 is.ULongArray && v2 is.ULongArray -> if (!v1.contentEquals(v2)) return false
else -> if (v1 != v2) return false
}

    }
    return true
}

@SinceKotlin("1.3")
@PublishedApi
@kotlin.jvm.JvmName("contentDeepToString")
@kotlin.js.JsName("contentDeepToStringImpl")
internal fun <T> Array<out T>.contentDeepToStringImpl(): String {
val length = size.coerceAtMost((Int.MAX_VALUE - 2) / 5) * 5 + 2 // in order not to overflow Int.MAX_VALUE
return buildString(length) {
contentDeepToStringInternal(this, mutableListOf())
}
}

@UseExperimental(ExperimentalUnsignedTypes::class)
private fun <T> Array<out T>.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<Array<*>>) {
if (this in processed) {
    result.append("[..."]
    return
}
processed.add(this)
result.append('[')

for (i in indices) {
    if (i != 0) {
        result.append(', '
    }

    val element = this[i]
    when (element) {
        null -> result.append("null")
        is Array<*> -> element.contentDeepToStringInternal(result, processed)
        is ByteArray -> result.append(element.contentToString())
        is ShortArray -> result.append(element.contentToString())
        is IntArray -> result.append(element.contentToString())
        is LongArray -> result.append(element.contentToString())
        is FloatArray -> result.append(element.contentToString())
        is DoubleArray -> result.append(element.contentToString())
        is CharArray -> result.append(element.contentToString())
        is BooleanArray -> result.append(element.contentToString())
        is UByteArray -> result.append(element.contentToString())
        is UShortArray -> result.append(element.contentToString())
        is UIntArray -> result.append(element.contentToString())
        is ULongArray -> result.append(element.contentToString())
        else -> result.append(element.toString())
    }

    result.append(']')
}
processed.removeAt(processed.lastIndex)
}
operator fun <V, V1 : V> MutableMap<String, out V1>.getValue(thisRef: Any?, property: KProperty<*>) =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V1)

@Deprecated("Use getValue() with two type parameters instead", level = DeprecationLevel.ERROR)
@kotlin.jvm.JvmName("getVarContravariant")
@kotlin.internal.LowPriorityInOverloadResolution
@kotlin.internal.InlineOnly
public inline fun <V> MutableMap<String, in V>.getValue(thisRef: Any?, property: KProperty<*>): V =
    @Suppress("UNCHECKED_CAST") (getOrImplicitDefault(property.name) as V)

/**
 * Stores the value of the property for the given object in this
 * mutable map.
 * @param thisRef the object for which the value is requested (not used).
 * @param property the property the metadata for the property, used to get the name of property and store the value associated with that name in the
 * map.
 * @param value the value to set.
 * @return the updated map.
 */
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<String, V>.setValue(thisRef: Any?, property: KProperty<*>, value: V) {
    this.put(property.name, value)
}

/**
 * Returns the value for the given key, or the implicit default value for this map.
 * By
 * default no implicit value is provided for maps and a [NoSuchElementException] is thrown.
 * To create a map with implicit default value use [withDefault] method.
 * @param key the key.
 * @return the value or the implicit default value.
 */
@kotlin.jvm.JvmName("getOrImplicitDefaultNullable")
@PublishedApi
internal fun <K, V> Map<K, V>.getOrImplicitDefault(key: K): V {
    if (this is MapWithDefault)
        return this.getOrImplicitDefault(key)
    return getOrElseNullable(key, { throw NoSuchElementException("Key $key is missing in the map.") })
}

/**
 * Returns a wrapper of this read-only map, having the implicit default value
 * provided with the specified function [defaultValue].
 *
 * This implicit default value is used when the original map doesn't contain a value for the key specified
 * and a value is obtained with [Map.getValue] function, for example when properties are delegated to the map.
 * When this map already has an implicit default value provided with a former call to [withDefault], it is being replaced by this call.
 * @param defaultValue the function that provides the implicit default value.
 * @return the updated map.
 */
public fun <K, V> Map<K, V>.withDefault(defaultValue: (key: K) -> V): Map<K, V> =
    when (this) {
        is MapWithDefault ->
            this.map.withDefault(defaultValue)
        else -> MapWithDefaultImpl(this, defaultValue)
    }

/**
 * Returns a wrapper of this mutable map, having the implicit default value provided with the specified function
 * [defaultValue].
 * @param defaultValue the function that provides the implicit default value.
 * @return the updated map.
 */
@kotlin.jvm.JvmName("withDefaultMutable")
public fun <K, V> MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =
    when (this) {
        is MutableMapWithDefault ->
            this.map.withDefault(defaultValue)
        else -> MutableMapWithDefaultImpl(this, defaultValue)
    }

private interface MapWithDefault<K, out V> : Map<K, V> {
    public val map: Map<K, V>
    public fun getOrImplicitDefault(key: K): V
}

private interface MutableMapWithDefault<K, V> : MutableMap<K, V>, MapWithDefault<K, V> {
    public override val map: MutableMap<K, V>
}

private class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>, private val default: (key: K) -> V) : MapWithDefault<K, V> {
    // implementation...
}

private class MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K, V>, private val default: (key: K) -> V) : MutableMapWithDefault<K, V> {
    // implementation...
}
V}, private val default: (key: K) -> V) : MutableMapWithDefault<K, V> {

    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()

    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)
    override fun get(key: K): V? = map.get(key)

    override val keys: MutableSet<K> get() = map.keys
    override val values: MutableCollection<V> get() = map.values
    override val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() = map.entries

    override fun put(key: K, value: V): V? = map.put(key, value)
    override fun remove(key: K): V? = map.remove(key)
    override fun putAll(from: Map<out K, V>) = map.putAll(from)
    override fun clear() = map.clear()

    override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.Random

/**
 * Removes a single instance of the specified element from this collection, if it is present.
 *<p>
 * Allows to overcome type-safety restriction of `remove` that requires to pass an element of type `E`:
 *<p>
 * @return `true` if the element has been successfully removed; `false` if it was not present in the collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)

/**
 * Removes all of this collection's elements that are also contained in the specified collection.
 *<p>
 * Allows to overcome type-safety restriction of `removeAll` that requires to pass a collection of type `Collection<E>`:
 *<p>
 * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<T>.removeAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).removeAll(elements)

/**
 * Retains only the elements in this collection that are contained in the specified collection.
 *<p>
 * @return `true` if any element was removed from the collection, `false` if the collection was not modified.
 */
@kotlin.internal.InlineOnly
public inline fun <T> MutableCollection<T>.retainAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).retainAll(elements)

/**
 * Adds the specified elemental to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(element: T) {
    this.add(element)
}

/**
 * Adds all elements of the given collections to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Iterable<T>) {
    this.addAll(elements)
}

/**
 * Adds all elements of the given array to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Array<T>) {
    this.addAll(elements)
}

/**
 * Adds all elements of the given sequence to this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.plusAssign(elements: Sequence<T>) {
    this.addAll(elements)
}

/**
 * Removes a single instance of the specified element from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(element: T) {
    this.remove(element)
}

/**
 * Removes all elements contained in the given collection from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Collection<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given array from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given sequence from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given elements array from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Array<T>) {
    this.removeAll(elements)
}

/**
 * Removes all elements contained in the given elements sequence from this mutable collection.
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableCollection<in T>.minusAssign(elements: Sequence<T>) {
    this.removeAll(elements)
}

*/
all elements of the given [elements] collection to this [MutableCollection].

public fun <T> MutableCollection<in T>.addAll(elements: Iterable<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (add(item)) result = true
    }
    return result
}

public fun <T> MutableCollection<in T>.addAll(elements: Sequence<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (add(item)) result = true
    }
    return result
}

public fun <T> MutableList<T>.removeFirst(): T?

private fun MutableList<*>.removeLast(): T?
if (isEmpty()) null else removeAt(0)

/**
 * Removes the last element from this mutable list and returns that
 * removed element, or throws [NoSuchElementException] if this list is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> MutableList<T>.removeLast(): T = if (isEmpty()) throw NoSuchElementException("List is empty.") else removeAt(lastIndex)

/**
 * Removes the last element from this mutable list and returns that removed element, or returns `null` if this list is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> MutableList<T>.removeLastOrNull(): T? = if (isEmpty()) null else removeAt(lastIndex)

/**
 * Removes all elements from this [MutableList] that match the given [predicate].
 * @return `true` if any element was removed from this collection, or `false` when no elements were removed and collection was not modified.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> MutableList<T>.removeAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, true)

/**
 * Retains only elements of this [MutableList] that match the given [predicate].
 * @return `true` if any element was removed from this collection, or `false` when all elements were retained and collection was not modified.
 */
public fun <T> MutableList<T>.retainAll(predicate: (T) -> Boolean): Boolean = filterInPlace(predicate, false)

private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    if (this !is RandomAccess)
        return (this as MutableList<T>).filterInPlace(predicate, predicateResultToRemove)

    var writeIndex: Int = 0
    for (readIndex in 0..lastIndex) {
        val element = this[readIndex]
        if (predicate(element) == predicateResultToRemove)
            continue

        if (writeIndex != readIndex)
            this[writeIndex] = element

        writeIndex++
    }
    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo writeIndex)
            removeAt(removeIndex)

        return true
    } else {
        return false
    }
}

/**
 * Randomly shuffles elements in this mutable list using the specified [random] instance as the source of randomness.
 */
@SinceKotlin("1.3")
public fun <T> MutableList<T>.shuffle(random: Random): Unit {
    for (i in lastIndex downTo 1) {
        val j = random.nextInt(i + 1)
        val copy = this[i]
        this[i] = this[j]
        this[j] = copy
    }
}

/**
 * Returns a reversed read-only view of the original List.
 * All changes made in the original list will be reflected in the reversed one.
 */
public fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this)

/**
 * Returns a reversed mutable view of the original mutable List.
 * All changes made in the original list will be reflected in the reversed one and vice versa.
 */
@kotlin.jvm.JvmName("asReversedMutable")
public fun <T> MutableList<T>.asReversed(): MutableList<T> = ReversedList(this)

// Source code is governed by the Apache 2.0 license.
```kotlin
import kotlin.sequences
import kotlin.coroutines.*
import kotlin.coroutines.intrinsics.*
import kotlin.experimental.ExperimentalTypeInference

/**
 * Builds a [Sequence] lazily yielding values one by one.
 * @see kotlin.sequences.generateSequence
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildFibonacciSequence
 */
@SinceKotlin("1.3")
public fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(block)
}

@SinceKotlin("1.3")
@Deprecated("Use 'sequence { }' function instead.", level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(builderAction)
}

/**
 * Builds an [Iterator] lazily yielding values one by one.
 * @sample samples.collections.Sequences.Building.buildIterator
 * @sample samples.collections.Iterables.Building.iterable
 */
@SinceKotlin("1.3")
public fun <T> iterator(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Iterator<T> {
    return 
}

@SinceKotlin("1.3")
@Deprecated("Use 'iterator { }' function instead.", level = DeprecationLevel.ERROR)
@kotlin.internal.InlineOnly
public inline fun <T> buildIterator(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Iterator<T> = iterator(builderAction)

/**
 * The scope for yielding values of a [Sequence] or an [Iterator],
 * provides [yield] and [yieldAll] suspension functions.
 * @see sequence
 * @see iterator
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Sequences.Building.buildFibonacciSequence
 */
@RestrictsSuspension
@SinceKotlin("1.3")
public abstract class SequenceScope<in T> internal constructor() {
    /**
     * Yields a value to the [Iterator] being built and suspends until the next value is requested.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public abstract suspend fun yield(value: T)

    /**
     * Yields all values from the `iterator` to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public abstract suspend fun yieldAll(iterator: Iterator<T>)

    /**
     * Yields potentially infinite sequence of values to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public suspend fun yieldAll(sequence: Sequence<T>)
}
```

State_ManyReady -> return true
else -> throw exceptionalState()}
|

@Suppress("UNCHECKED_CAST")
val result = nextValue as T
nextValue = null

override fun next(): T {
when (state) {
State_NotReady, State_ManyNotReady -> return nextNotReady()
State_ManyReady -> {
state = State_ManyNotReady
return nextIterator!!().next()
}
State_Ready -> {
state = State_NotReady

buffer.size

private fun nextNotReady(): T {
if (!hasNext()) throw NoSuchElementException() else return next()}

private fun exceptionalState(): Throwable = when (state) {
State_Done -> NoSuchElementException()
State_Failed -> IllegalStateException("Iterator has failed.")
else -> IllegalStateException("Unexpected state of the iterator: $state")
}

override suspend fun yield(value: T) {
nextValue = value
state = State_Ready
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

override suspend fun yieldAll(iterator: Iterator<T>) {
if (!iterator.hasNext()) return

nextIterator = iterator
state = State_NotReady
return suspendCoroutinesUninterceptedOrReturn {
c ->
nextStep = c
COROUTINE_SUSPENDED
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}

// Completion continuation implementation
override fun resumeWith(result: Result<Unit>) {
result.getOrThrow() // just rethrow exception if it is there
state = State_Done

override val coroutineContext: CoroutineContext
get() = EmptyCoroutineContext
}
private val capacity = buffer.size
private var startIndex: Int = 0
override fun get(index: Int): T {
    @Suppress("UNCHECKED_CAST")
    return buffer[ startIndex.forward(index) ] as T
}

override fun set(index: Int, value: T) {
    @Suppress("UNCHECKED_CAST")
    buffer[ startIndex.forward(index) ] = value
}

override fun contains(element: T): Boolean {
    return buffer.contains(element)
}

override fun removeFirst(n: Int) {
    if (n <= 0) return
    val start = startIndex
    val end = start.forward(n)
    if (start > end) {
        buffer.fill(null, start, capacity)
        buffer.fill(null, 0, end)
    } else {
        buffer.fill(null, start, end)
    }
    startIndex = end
    size -= n
}

override fun removeLast(n: Int) {
    if (n <= 0) return
    val start = startIndex
    val end = start.forward(size - n - 1)
    if (start > end) {
        buffer.fill(null, start, capacity)
        buffer.fill(null, 0, end)
    } else {
        buffer.fill(null, start, end)
    }
    size -= n
}

override fun clear() {
    buffer.fill(null, 0, size)
    startIndex = 0
    size = 0
}

override fun size(): Int {
    return size
}

override fun isFull(): Boolean {
    return size == capacity
}

override fun iterator(): Iterator<T> = object : AbstractIterator<T>() {
    private var count = size
    private var index = startIndex

    override fun computeNext() {
        if (count == 0) {
            done()
        } else {
            @Suppress("UNCHECKED_CAST")
            setNext(buffer[index] as T)
            index = index.forward(1)
            count--
        }
    }
}

override fun <T> toArray(array: Array<T>): Array<T> {
    val result: Array<T?> = if (array.size < this.size) array.copyOf(this.size) else array as Array<T?>
    val size = this.size
    var widx = 0
    var i = startIndex
    while (widx < size && i < capacity) {
        result[widx] = buffer[i] as T
        widx++
        i++
    }
    i = 0
    while (widx < size) {
        result[widx] = buffer[i] as T
        widx++
        i++
    }
    if (result.size > this.size) result[size] = null
    return result
}

fun <T> expanded(capacity: Int): RingBuffer<T> {
    val newCapacity = (capacity + (capacity shr 1) + 1).coerceAtMost(maxCapacity)
    val newBuffer = if (startIndex == 0) buffer.copyOf(newCapacity) else toArray(arrayOfNulls(newCapacity))
    return RingBuffer(newBuffer, size)
}

fun add(element: T) {
    if (isFull()) {
        throw IllegalStateException("ring buffer is full")
    }
    buffer[startIndex.forward(size)] = element
    size++
}

fun removeFirst(n: Int) {
    require(n >= 0) { "n shouldn't be negative but it is $n" }
    require(n <= size) { "n shouldn't be greater than the buffer size: n = $n, size = $size" }
    if (n > 0) {
        val start = startIndex
        val end = start.forward(n)
        if (start > end) {
            buffer.fill(null, start, capacity)
            buffer.fill(null, 0, end)
        } else {
            buffer.fill(null, start, end)
        }
        startIndex = end
        size -= n
    }
}

private fun Int.forward(n: Int): Int = (this + n) % capacity

@Suppress("NOTHING_TO_INLINE")
in private fun partition(array: UByteArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    var pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@Suppress("NOTHING_TO_INLINE")
in private fun quickSort(array: UByteArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1) quickSort(array, left, index - 1)
    if (index < right) quickSort(array, index, right)
}

@Suppress("NOTHING_TO_INLINE")
in private fun partition(array: UShortArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    var pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@Suppress("NOTHING_TO_INLINE")
in private fun quickSort(array: UShortArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1) quickSort(array, left, index - 1)
    if (index < right) quickSort(array, index, right)
}

@Suppress("NOTHING_TO_INLINE")
in private fun partition(array: UIntArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    var pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@Suppress("NOTHING_TO_INLINE")
in private fun quickSort(array: UIntArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1) quickSort(array, left, index - 1)
    if (index < right) quickSort(array, index, right)
}
var j = right
val pivot = array[(left + right) / 2]
while (i <= j) {
    while (array[i] < pivot) i++
    while (array[j] > pivot) j--
    if (i <= j) {
        val tmp = array[i]
        array[i] = array[j]
        array[j] = tmp
        i++
        j--
    }
}

@ExperimentalUnsignedTypes
private fun quickSort(array: UIntArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1) quickSort(array, left, index - 1)
    if (index < right) quickSort(array, index, right)
}

 Interfaces

="="*/

 Miscellaneous
*\n  A function declaring the `callsInPlace` effect must be _inline_. */
    /*
    * @sample
    */
    @ContractsDsl public fun <R> callsInPlace(lambda: Function<R>, kind: InvocationKind = InvocationKind.UNKNOWN): CallsInPlace
}

/**
 * Specifies how many times a function invokes its function parameter in place.
 * See [ContractBuilder.callsInPlace] for the details of the call-in-place function contract.
 */
@ContractsDsl @ExperimentalContracts @SinceKotlin("1.3")
public enum class InvocationKind {
    /**
     * A function parameter will be invoked one time or not invoked at all.
     */
    AT_MOST_ONCE,

    /**
     * A function parameter will be invoked exactly one time.
     */
    EXACTLY_ONCE,

    /**
     * A function parameter is called in place, but it’s unknown how many times it can be called.
     */
    UNKNOWN
}

/**
 * Specifies the contract of a function.
 *
 * The contract description must be at the beginning of a function and have at least one
 * effect.
 *
 * Only the top-level functions can have a contract for now.
 *
 * @param builder the lambda where
 * the contract of a function is described with the help of the [ContractBuilder] members.
 */
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package kotlin.coroutines

/**
 * Marks coroutine context element that
 * intercepts coroutine continuations.
 * The coroutines framework uses [ContinuationInterceptor.Key] to retrieve the
 * interceptor and
 * intercepts all coroutine continuations with [interceptContinuation] invocations.
 * [ContinuationInterceptor] behaves like a [polymorphic element][AbstractCoroutineContextKey], meaning that
 * its implementation delegates [get][CoroutineContext.Element.get] and
 * [minusKey][CoroutineContext.Element.minusKey] to
 * [getPolymorphicElement] and [minusPolymorphicKey] respectively.
 * [ContinuationInterceptor] subtypes can be extracted from the coroutine context using either
 * [ContinuationInterceptor.Key] or subtype key if it extends [AbstractCoroutineContextKey].
 * The key that defines *the* context interceptor.
 * Returns continuation that wraps the original
 * continuation, thus intercepting all resumptions.
 * This function is invoked by coroutines framework when
 * the resulting continuations are
 * cached internally per each instance of the original
 * continuation.
 * This function may simply return original
 * [continuation] if it does not want to intercept this particular
 * continuation.
 * When the original
 * [continuation] completes, coroutine framework invokes
 * [releaseInterceptedContinuation]
 * with the resulting continuation if it was intercepted, that is if
 * `interceptContinuation` had previously returned a different continuation instance.
 * Invoked for the
 * continuation instance returned by [interceptContinuation] when the original
 * continuation completes and will
not be used anymore. This function is invoked only if [interceptContinuation]
* had returned a different
continuation instance from the one it was invoked with.
* Default implementation does nothing.
* @param continuation Continuation instance returned by this interceptor's [interceptContinuation] invocation.
* @return null if nothing done by default.
* public fun releaseInterceptedContinuation(continuation: Continuation<?*>?)
* do nothing by default.
* E?
* // getPolymorphicKey specialized for ContinuationInterceptor key
*@OptIn(ExperimentalStdlibApi::class) if (key is AbstractCoroutineContextKey<*, *>)
{return null}
@Suppress("UNCHECKED_CAST")
return (key.isSubKey(this.key)) key.tryCast(this) as? E else null
@Suppress("UNCHECKED_CAST")
return if (ContinuationInterceptor === key) this as E else null
* public override fun minusKey(key: CoroutineContext.Key<*>):
CoroutineContext // minusPolymorphicKey specialized for ContinuationInterceptor key
*@OptIn(ExperimentalStdlibApi::class)
if (key is AbstractCoroutineContextKey<*, *>)
return if (key.isSubKey(this.key) &&
key.tryCast(this) != null) EmptyCoroutineContext else this
else
    return if
(ContinuationInterceptor === key) EmptyCoroutineContext else this

/**
 * Persistent context for the coroutine. It is an indexed set of 
[Element] instances.
* An indexed set is a mix between a set and a map.
* Every element in this set has a unique [Key].
* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
*/

package kotlin.coroutines

/**
 * Base class for 
* Base class for [CoroutineContext.Key] associated with polymorphic
 */
public abstract class AbstractCoroutineContextElement(public override val key:
Key<*>) : Element

/**
 * Base class for 
[CoroutineContext.Key] associated with polymorphic
 */
public abstract class AbstractCoroutineContextKey

[get][Element.get] and [minusKey][Element.minusKey] in * to [getPolymorphicElement] and [minusPolymorphicKey] respectively. * in * Polymorphic elements can be extracted from the coroutine context using both element key and its supertype key. * Example of polymorphic elements: in * ```
BaseElement : CoroutineContext.Element

override val key: CoroutineContext.Key<*>

get() = Key

// It is important to use getPolymorphicKey and minusPolymorphicKey


E? = getPolymorphicElement(key)

override fun minusKey(key: CoroutineContext.Key<*>):

CoroutineContext = minusPolymorphicKey(key)

* in * class DerivedElement : BaseElement

companion object Key : AbstractCoroutineContextKey<BaseElement, DerivedElement>(BaseElement, { it as?

DerivedElement })

* in * // Now it is possible to query both `BaseElement` and `DerivedElement`

someContext[BaseElement] // Returns BaseElement?, non-null both for BaseElement and DerivedElement

someContext[DerivedElement] // Returns DerivedElement?, non-null only for DerivedElement

```
public override fun minusKey(key: Key<*>):
    CoroutineContext {
        element[key]?.let { return left } if
        newLeft = left.minusKey(key)
        return
    }

    public override fun minusKey(key: Key<*>):
        CoroutineContext {
            element[key]?.let { return left }
            val newLeft = left.minusKey(key)
            return
        }

        public override fun override fun equals(other: Any?): Boolean =
            this === other || other is CombinedContext &&
            other.size() == size &&
            other.containsAll(this)

        override fun hashCode(): Int = left.hashCode() + element.hashCode()

        override fun toString(): String =
            "[" + fold(Unit) { acc, element ->
                if (acc.isEmpty()) element.toString() else
                "$acc, $element"
            } + "]"

        private fun writeReplace(): Any {
            val n = size()
            val elements =
                arrayOfNulls<CoroutineContext>(n)
            var index = 0
            fold(Unit) { _, element -> elements[index++] =
                element }
            check(index == n)
            @Suppress("UNCHECKED_CAST")
            return Serialized(elements as Array<CoroutineContext>)
        }
    }
}

@file:kotlin.jvm.JvmName("IntrinsicsKt")
@file:kotlin.jvm.JvmMultifileClass

package
    kotlin.coroutines.intrinsics

import kotlin.coroutines.*
import kotlin.internal.InlineOnly

/**
 * Obtains the
 * current continuation instance inside suspend functions and either suspends
 * currently running coroutine or returns
 * result immediately without suspension.
 * If the [block] returns the special [COROUTINE_SUSPENDED]
 * value, it means that suspend function did suspend the execution and will
 * not return any result immediately. In
 * this case, the [Continuation] provided to the [block] shall be resumed by invoking [Continuation.resumeWith] at
 * some moment in the
 * future when the result becomes available to resume the computation.
 * Otherwise, the return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.
 * It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.
 * As the result type of the [block] is declared as `Any?` and cannot be correctly type-checked,
 * its proper return type remains on the conscience of the suspend function's author.
 * [Invocation of [Continuation.resumeWith]] resumes coroutine directly in the invoker's thread without going through the
 * [Continuation] that might be present in the coroutine's
 * [CoroutineContext].
 * It is the invoker's responsibility to ensure that a proper invocation context is established.
 * [Continuation.intercepted] can be used to acquire the intercepted continuation.
 * Note that it is not recommended to call either [Continuation.resume] or
 * [Continuation.resumeWithException] functions synchronously in the
 * same stackframe where suspension function is run. Use [suspendCoroutine] as a safer way to obtain current
 * continuation instance.
 */
@SinceKotlin("1.3")
@InlineOnly
@Suppress("UNUSED_PARAMETER",
    "RedundantSuspendModifier")
public suspend inline fun <T> suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T
    throw
    @file:kotlin.jvm.JvmName("IntrinsicsKt")
    @file:kotlin.jvm.JvmMultifileClass
    package
kotlin.coroutines.intrinsics

import kotlin.coroutines.*

private class Serialized(val elements: Array<CoroutineContext>) : Serializable {
    companion object {
        private const val serialVersionUID: Long = 0L
    }

    private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)
}

private val COROUTINE_SUSPENDED: Any get() = CoroutineSingletons.COROUTINE_SUSPENDED

private fun contains(element: Element): Boolean =
    get(element.key) == element

private fun containsAll(context: CombinedContext): Boolean {
    var cur = context
    while (true) {
        if (!contains(cur.element)) return false
        val next = cur.left
        if (next is CombinedContext) {
            cur = next
        } else {
            return contains(next as Element)
        }
    }
}

override fun equals(other: Any?): Boolean =
    this === other || other is CombinedContext &&
    other.size() == size() &&
    other.containsAll(this)

override fun hashCode(): Int = left.hashCode() + element.hashCode()

override fun toString(): String =
    "[" + fold("") { acc, element ->
        if (acc.isEmpty()) element.toString() else
        "$acc, $element"
    } + "]"

private fun writeReplace(): Any {
    val n = size()
    val elements =
        arrayOfNulls<CoroutineContext>(n)
    var index = 0
    fold(Unit) { _, element -> elements[index++] =
        element }
    check(index == n)
    @Suppress("UNCHECKED_CAST")
    return Serialized(elements as Array<CoroutineContext>)
}

private class Serialized(val elements: Array<CoroutineContext>) : Serializable {
    companion object {
        private const val serialVersionUID: Long = 0L
    }

    private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)
}
here ensures two important properties: 1. It makes SafeContinuation serializable with all kinds of serialization frameworks (since all of them natively support enums) 2. It improves debugging experience, since you clearly see toString() value of those objects and what package they come from:

```kotlin
@SinceKotlin("1.3")
@PublishedApi // This class is Published API via serialized representation of SafeContinuation, don't rename/move
internal enum class CoroutineSingletons { COROUTINE_SUSPENDED, UNDECIDED, RESUMED } /* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

package kotlin.experimental

/** Performs a bitwise AND operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.and(other: Byte): Byte = (this.toInt() and other.toInt()).toByte()

/** Performs a bitwise OR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.or(other: Byte): Byte = (this.toInt() or other.toInt()).toByte()

/** Performs a bitwise XOR operation between the two values. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.xor(other: Byte): Byte = (this.toInt() xor other.toInt()).toByte()

/** Inverts the bits in this value. */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Byte.inv(): Byte = (this.toInt().inv()).toByte()

// Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

package kotlin.internal

/** The experimental marker for type inference augmenting annotations. */
@Suppress("DEPRECATION")
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.ANNOTATION_CLASS)
@SinceKotlin("1.3")
public annotation class ExperimentalTypeInference

/** Specifies that the constraint built for the type during type inference should be an equality one. */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@Retention(AnnotationRetention.BINARY)
@MustBeDocumented
@MustBeDocumented
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.ANNOTATION_CLASS)
@Supress("DEPRECATION")
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(AnnotationTarget.ANNOTATION_CLASS)
@ExperimentalTypeInference

/** Specifies that the corresponding type should be ignored during type inference */
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
@inference annotation class NoInfer

/** Specifies that the constraint built for the type during type inference should be an equality one. */
@Target(AnnotationTarget.TYPE)
@Retention(AnnotationRetention.BINARY)
@inference annotation class Exact

/** Specifies that a corresponding member has the lowest priority in overload resolution. */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
@inference annotation class LowPriorityInOverloadResolution

/** Specifies that the corresponding member has the highest priority in overload resolution. Effectively this means that the new extension annotated with this annotation will win in overload resolution over a member with the same signature. */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
@inference annotation class HidesMembers

* The value of this type parameter should be mentioned in input types (argument types,
receiver type or expected type).

*/
@Target(AnnotationTarget.TYPE_PARAMETER)
@Retention(AnnotationRetention.BINARY)
internal annotation class OnlyInputTypes

/**
 * Specifies that this function should not be called directly without
 * inlining
 */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY,
AnnotationTarget.PROPERTY_GETTER,
AnnotationTarget.PROPERTY_SETTER)
@Retention(AnnotationRetention.BINARY)
internal annotation class InlineOnly

/**
 * The value of this parameter should be a property reference expression (`this::foo`),
 * referencing a `lateinit` property.
 */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY,
AnnotationTarget.PROPERTY_GETTER)
@Retention(AnnotationRetention.BINARY)
internal annotation class DynamicExtension

/**
 * Specifies that this declaration can have dynamic receiver type.
 */
@Target(AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY)
@Retention(AnnotationRetention.BINARY)
internal annotation class DynamicExtension

/**
 * Specifies that this parameter should be compared with the [version] value, when compiling the usage of the annotated
 * declaration.
 * Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported,
 * so the Kotlin compiler before 1.2 is going to treat any [RequireKotlin] as if it requires the language version.
 * Since 1.2, the Kotlin compiler supports
 * [RequireKotlinVersionKind.LANGUAGE_VERSION], [RequireKotlinVersionKind.COMpiler_VERSION] and [RequireKotlinVersionKind.API_VERSION].
 * If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin),
 * Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a
 * requirement on the language version.
 * This annotation is erased at compile time; its arguments are stored
 * in a more compact form in the Kotlin metadata.
 */
@Target(AnnotationTarget.CLASS,
AnnotationTarget.FUNCTION,
AnnotationTarget.PROPERTY,
AnnotationTarget.CONSTRUCTOR,
AnnotationTarget.TYPEalias)
@Retention(AnnotationRetention.SOURCE)
@Repeatable
@SinceKotlin("1.2")
internal annotation class RequireKotlin(
    val version: String,
    val message: String = "",
    val level: DeprecationLevel = DeprecationLevel.ERROR,
    val versionKind: RequireKotlinVersionKind =
    RequireKotlinVersionKind.LANGUAGE_VERSION,
    val errorCode: Int = -1
)

/**
 * The kind of the
 * version that is required by [RequireKotlin].
 */
@SinceKotlin("1.2")
internal enum class RequireKotlinVersionKind {
    LANGUAGE_VERSION,
    COMPILER_VERSION,
    API_VERSION,
}

/**
 * Specifies that this declaration is a part of special DSL, used for constructing
 * function's contract.
 */
@Retention(AnnotationRetention.RUNTIME)
@SinceKotlin("1.2")
internal annotation class ContractsDsl

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Standard
 * property delegates.
 */
public object Delegates {
    /**
     * Returns a property delegate for a read/write
     * property with a non-`null` value that is initialized not during
     * object construction time but at a later time.
     * Trying to read the property before the initial value has been
     * assigned results in an exception.
     */
    public fun <T : Any> notNull(): ReadWriteProperty<T, T> = NotNullVar()

    /**
     * Returns a property delegate for a read/write
     * property that calls a specified callback function when changed.
     */
    public fun <T : Any> onChange: onChange: (value: T) -> Unit = { param -> }
samples.properties.Delegates.observableDelegate

    public inline fun <T> observable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Unit):ReadWriteProperty<Any?, T> =
        object : ObservableProperty<T>(initialValue) {
            override fun afterChange(property: KProperty<*>, oldValue: T, newValue: T) = onChange(property, oldValue, newValue)

        }

    public inline fun <T> vetoable(initialValue: T, crossinline onChange: (property: KProperty<*>, oldValue: T, newValue: T) -> Boolean):
        ReadWriteProperty<Any?, T> =
        object : ObservableProperty<T>(initialValue) {
            override fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = onChange(property, oldValue, newValue)
        }

private class NotNullVar<T : Any>() : ReadWriteProperty<Any?, T> {
    private var value: T? = null

    public override fun getValue(thisRef: Any?, property: KProperty<*>): T {
        return value ?: throw IllegalStateException("Property ${property.name} should be initialized before get.")
    }

    public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {
        this.value = value
    }
}

package kotlin.properties

import kotlin.reflect.KProperty

/**
 * Implements the core logic of a property delegate for a read/write property that calls callback functions when changed.
 * @param initialValue the initial value of the property.
 */
public abstract class ObservableProperty<T>(initialValue: T) :
        ReadWriteProperty<Any?, T> {
    private var value = initialValue

    /**
     * The callback which is called after the change of the property is made. The value of the property has already been changed when this callback is invoked.
     * If the callback returns `true` the value of the property is being set to the new value, and if the callback returns `false` the new value is discarded and the property remains its old value.
     */
    protected open fun afterChange(property: KProperty<*>, oldValue: T, newValue: T): Unit {}

    public override fun getValue(thisRef: Any?, property: KProperty<*>): T {
        return value
    }

    public override fun setValue(thisRef: Any?, property: KProperty<*>, value: T) {
        val oldValue = this.value
        if (!beforeChange(property, oldValue, value)) {
            return
        }

        this.value = value
        afterChange(property, oldValue, value)
    }
}

package kotlin.random

import kotlin.math.nextDown

/**
 * An abstract class that is implemented by random number generator algorithms.
 * The companion object [Random.Default] is the default instance of [Random].
 * Get a seeded instance of random generator use [Random] function.
 * @sample samples.random.Randoms.defaultRandom
 */
@SinceKotlin("1.3")
public abstract class Random {

    /**
     * Gets the next random `Int` number of bits.
     * Generates an `Int` whose lower [bitCount] bits are filled with random values and the remaining upper bits are zero.
     * @param bitCount number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.
     * @param bitCount number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.
     * @param bitCount number of bits to generate, must be in range 0..32, otherwise the behavior is unspecified.
     */
    public fun nextInt(): Int = nextBits(32)

    /**
     * Gets the next random `Int` from the random number generator.
     * Generates an `Int` random value uniformly distributed between `Int.MIN_VALUE` and `Int.MAX_VALUE` (inclusive).
     */
    public fun random(): Int = nextInt()
(inclusive) and the specified [until] bound (exclusive).

*param must be positive.*

*public open fun nextInt(until: Int): Int =

nextInt(0, until)/**/ * Gets the next random 'Int' from the random number generator in the specified range.*

*generates an 'Int' random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.*

*public open fun nextInt(from: Int, until: Int): Int =

checkRangeBounds(from, until)

val n = until - from

if (n > 0) {

val val bitCount = fastLog2(n)

nextBits(bitCount)

} else {

var v: Int

do {

val bits = nextInt().ushr(1)

v = bits % n

} while (bits - v + (n - 1) < 0)

v

}*/

*Generates a 'Long' random value uniformly distributed between 'Long.MIN_VALUE' and 'Long.MAX_VALUE' (inclusive).*

*public open fun nextLong(): Long =

nextLong(0, until)/**/ * Gets the next random non-negative 'Long' from the random number generator less than the specified [until] bound.*

*Generates a 'Long' random value uniformly distributed between '0' (inclusive) and the specified [until] bound (exclusive).*

*param must be positive.*

*throws IllegalArgument exception if [until] is negative or zero.*

*public open fun nextLong(until: Long) =

nextLong(0, until)/**/ * Gets the next random 'Long' from the random number generator in the specified range.*

*Generates a 'Long' random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.*

*@throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextLong(from: Long, until: Long): Long =

checkRangeBounds(from, until)

val n = until - from

if (n > 0) {

val val rnd = nextInt()

if (rnd in from until until) return rnd

} else {

while (true) {

val rnd = nextInt()

if (rnd in from until until) return rnd

}*/

*Generates a 'Double' random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).*

*Generates a 'Double' random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.*

*throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextDouble(): Double =

doubleFromParts(nextBits(26), nextBits(27))

*Generates a 'Double' random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.*

*@throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextDouble(until: Double): Double =

doubleFromParts(nextBits(26), nextBits(27))

*Generates a 'Double' random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.*

*throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextDouble(until: Double): Double =
nextDouble(0.0, until)/**/ * Gets the next random 'Double' from the random number generator less than the specified [until] bound.*

*Generates a 'Double' random value uniformly distributed between 0 (inclusive) and 1 (exclusive).*

*throws IllegalArgumentException if [until] is negative or zero.*

*public open fun nextDouble(): Double =
doubleFromParts(nextBits(26), nextBits(27))

*Generates a 'Double' random value uniformly distributed between 0 (inclusive) and 1 (exclusive).*

*throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextDouble(from: Double, until: Double): Double =

checkRangeBounds(from, until)

val size = until - from

val r = if (size.isInfinite()) && from.isFinite() && until.isFinite())

val r = if (size.isInfinite())

val r = from + nextDouble() * size

if (r >= until)

nextDown() else r

}*/

*Generates a 'Double' random value uniformly distributed between 0 (inclusive) and 1 (exclusive).*

*throws IllegalArgumentException if [from] is greater than or equal to [until].*

*public open fun nextDouble(): Double =
nextDouble(24) / (1 shl 24).toFloat()/**/ * Fills a subrange of the specified byte array starting from [fromIndex] inclusive and ending [toIndex] exclusive

*with random bytes.*

*throws IllegalArgumentException if [from] is greater than or equal to [until].*
@return [array] with the subrange filled with random bytes.

@Override
public open fun nextBytes(array: ByteArray, fromIndex: Int = 0, toIndex: Int = array.size): ByteArray {
    require(fromIndex in 0..array.size && toIndex in 0..array.size) { "fromIndex ($fromIndex) or toIndex ($toIndex) are out of range: 0..${array.size}." }
    require(fromIndex <= toIndex) { "fromIndex ($fromIndex) must be not greater than toIndex ($toIndex)." }

    val steps = (toIndex - fromIndex) / 4
    var position = fromIndex
    repeat(steps) {
        val v = nextInt()
        array[position] = v.toByte()
        array[position + 1] = v.ushr(8).toByte()
        array[position + 2] = v.ushr(16).toByte()
        array[position + 3] = v.ushr(24).toByte()
        position += 4
    }

    val remainder = toIndex - position
    val vr = nextBits(remainder * 8)
    for (i in 0 until remainder) {
        array[position + i] = vr.ushr(i * 8).toByte()
    }

    return array
}

/**
 * Fills the specified byte [array] with random bytes and returns it.
 * @return [array] filled with random bytes.
 */
public open fun nextBytes(array: ByteArray): ByteArray = nextBytes(array, 0, array.size)

/**
 * Creates a byte array of the specified [size], filled with random bytes.
 */
public open fun nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))

/**
 * The default random number generator.
 * @sample samples.random.Randoms.defaultRandom
 */
companion object Default : Random() {
    private val defaultRandom: Random = defaultPlatformRandom()
    override fun nextBits(bitCount: Int): Int = defaultRandom.nextBits(bitCount)
    override fun nextInt(): Int = defaultRandom.nextInt()
    override fun nextInt(until: Int): Int = defaultRandom.nextInt(until)
    override fun nextInt(from: Int, until: Int): Int = defaultRandom.nextInt(from, until)
    override fun nextLong(): Long = defaultRandom.nextLong()
    override fun nextLong(until: Long): Long = defaultRandom.nextLong(until)
    override fun nextLong(from: Long, until: Long): Long = defaultRandom.nextLong(from, until)
    override fun nextBoolean(): Boolean = defaultRandom.nextBoolean()
    override fun nextDouble(): Double = defaultRandom.nextDouble()
    override fun nextDouble(until: Double): Double = defaultRandom.nextDouble(until)
    override fun nextDouble(from: Double, until: Double): Double = defaultRandom.nextDouble(from, until)
    override fun nextFloat(): Float = defaultRandom.nextFloat()
    override fun nextBytes(array: ByteArray): ByteArray = defaultRandom.nextBytes(array)
    override fun nextBytes(size: Int): ByteArray = defaultRandom.nextBytes(size)
    override fun nextBytes(array: ByteArray, fromIndex: Int, toIndex: Int): ByteArray = defaultRandom.nextBytes(array, fromIndex, toIndex)

    @Deprecated("Use Default companion object instead", level = DeprecationLevel.HIDDEN)
    @Suppress("DEPRECATION_ERROR")
    @kotlin.jvm.JvmField
}
}

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Int` value.
 * @sample samples.random.Randoms.seededRandom
 */
@SinceKotlin("1.3")
public fun Random(seed: Int): Random = XorWowRandom(seed, seed.shr(31))

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Long` value.
 * @sample samples.random.Randoms.seededRandom
 */
@SinceKotlin("1.3")
public fun Random(seed: Long): Random = XorWowRandomRandom(seed, seed.toLong())

/**
 * Gets the next random `Int` from the random number generator in the specified [range].
 * @throws IllegalArgumentException if [range] is empty.
 */
@SinceKotlin("1.3")
public fun Random.nextInt(range: IntRange): Int = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)
    else -> throw IllegalArgumentException("Cannot get random in range: $range")
}
range.first > Int.MIN_VALUE -> nextInt(range.first - 1, range.last) + 1 elif not range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1) else -> nextLong(range.first > Long.MIN_VALUE -> nextLong(range.first - 1, range.last) + 1 else -> nextLong(un过 internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) { boundsErrorMessage(from, until) } })

/** Takes upper [bitCount] bits (0..32) from this number. */
internal fun Int.takeUpperBits(bitCount: Int): Int =
    this.ushr(32 - bitCount) and (-bitCount).shr(31)

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    boundsErrorMessage(from, until)
}

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the apache/LICENSE.txt file.
 */

package kotlin.random

/**
 * Gets the next random [UInt] from the random number generator.
 *
 * Generates a [UInt] random value uniformly distributed between [UInt.MIN_VALUE] and [UInt.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt = nextInt().toUInt()

/**
 * Gets the next random [UInt] from the random number generator less than the specified [until] bound.
 *
 * Generates a [UInt] random value uniformly distributed between 0 (inclusive) and the specified [until] bound (exclusive).
 *
 * @throws IllegalArgumentException if [until] is zero.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.next UInt(until: UInt): UInt = nextUInt(0u, until)

/**
 * Gets the next random [UInt] from the random number generator in the specified range.
 *
 * Generates a [UInt] random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.
 *
 * @throws IllegalArgumentException if [from] is greater than or equal to [until].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.next UInt(from: UInt, until: UInt): UInt {
    checkUIntRangeBounds(from, until)

    val signedFrom = from.toInt() xor Int.MIN_VALUE
    val signedUntil = until.toInt() xor Int.MIN_VALUE

    val signedResult = nextInt(signedFrom, signedUntil) xor Int.MIN_VALUE
    return signedResult.toUInt()
}

/**
 * Gets the next random [ULong] from the random number generator.
 *
 * Generates a [ULong] random value uniformly distributed between [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(): ULong = nextLong().toULong()

/**
 * Gets the next random [ULong] from the random number generator less than the specified [until] bound.
 *
 * Generates a [ULong] random value uniformly distributed between `0` (inclusive) and the specified [until] bound (exclusive).
 *
 * @throws IllegalArgumentException if [until] is zero.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextULong(until: ULong): ULong = nextULong(0uL, until)

/**
 * Gets the next random [ULong] from the random number generator in the specified range.
 *
 * Generates a [ULong] random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.
 */

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the apache/LICENSE.txt file.
 */

package kotlin.random

/**
 * Gets the next random [Long] from the random number generator in the specified [range].
 *
 * Generates a 'Long' random value uniformly distributed in the specified [range]:
 */

@WithContinuation
public fun Random.nextLong(continuation: Continuation<Long>) =
    nextLong.nextLong(continuation)

/**
 * Gets the next random [Long] from the random number generator.
 *
 * Generates a 'Long' random value uniformly distributed between [Long.MIN_VALUE] and [Long.MAX_VALUE].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextLong(): Long = when {
    range.isEmpty() -> throw
    range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1)
    range.first > Long.MIN_VALUE -> nextLong(range.first - 1, range.last) + 1
    else -> nextLong()
}

/**
 * Gets the next random [Long] from the random number generator.
 *
 * Generates a 'Long' random value uniformly distributed between [Long.MIN_VALUE] and [Long.MAX_VALUE] (inclusive).
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextLong(range: LongRange): Long = when {
    range.isEmpty() -> throw
    range.last < Long.MAX_VALUE -> nextLong(range.first, range.last + 1)
    range.first > Long.MIN_VALUE -> nextLong(range.first - 1, range.last) + 1
    else -> nextLong()
}

@UseExperimental(ExperimentalStdlibApi::class)
internal fun fastLog2(value: Int): Int = 31 - value.countLeadingZeroBits()

/** Takes upper [bitCount] bits (0..32) from this number. */
internal fun Int.takeUpperBits(bitCount: Int): Int =
    this.ushr(32 - bitCount) and (-bitCount).shr(31)

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Double, until: Double) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun boundsErrorMessage(from: Any, until: Any) = "Random range is empty: [$from, $until).

Open Source Used In DNAC Platform Arica  13486
IllegalArgumentException if [from] is greater than or equal to [until].

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextULong(from: ULong, until: ULong): ULong {\n    checkULongRangeBounds(from, until) xor ULong.MIN_VALUE { val signedFrom = from.toLong() xor ULong.MIN_VALUE; val signedUntil = until.toLong() xor ULong.MIN_VALUE; return signedResult } toULong() \n\} /*\n* Generates a [ULong] random value uniformly distributed in the specified [range].\n*/

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextULong(range: ULongRange): ULong = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < ULong.MAX_VALUE -> nextULong(range.first, range.last + 1u)
    range.first > ULong.MIN_VALUE -> nextULong(range.first - 1u, range.last) + 1u
    else -> nextULong(\n\} /*\n* Fills the specified unsigned byte array with random bytes and returns it.\n*/

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextUBytes(array: UByteArray) {\n    nextBytes(array.asByteArray()) return array\n\} /*\n* Creates an unsigned byte array of the specified [size], filled with random bytes.\n*/

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextUBytes(size: Int): UByteArray = nextBytes(size).asUByteArray()

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextUBytes(array: UByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray {
    nextBytes(array.asByteArray(), fromIndex, toIndex)
    return array
\} /*\n* Fills a subrange of the specified 'UByteArray' array starting from [fromIndex] inclusive and ending [toIndex] exclusive with random UBytes.\n*/

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\public fun Random.nextBits(bitCount: Int): Int =
    nextInt().takeUpperBits(bitCount)

*\@ExperimentalUnsignedTypes\internal fun checkUIntRangeBounds(from: UInt, until: UInt) =
    require(until > from) { boundsErrorMessage(from, until) }

*\@ExperimentalUnsignedTypes\internal fun checkULongRangeBounds(from: ULong, until: ULong) =
    require(until > from) { boundsErrorMessage(from, until) }

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

/**
 * Represents a range of \[Comparable\] values.
 */
private open class ComparableRange<T : Comparable<T>>(
    override val start: T,
    override val endInclusive: T
) : ClosedRange<T> {

    override fun equals(other: Any?): Boolean {
        return other is ComparableRange<*> && (isEmpty() && other.isEmpty() ||
                start == other.start && endInclusive == other.endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else start.hashCode() + endInclusive.hashCode()
    }

    override fun

this class.

* This is an experimental function that behaves as a similar function from kotlin.reflect.full on JVM.

* @n * @see [KClass.isInstance]

* @n * @see [KClass.safeCast]

* @n @SinceKotlin("1.3")@n @ExperimentalStdlibApi

fun <T : Any> KClass<T>.cast(value: Any?): T {
    return value as T
}

/**
 * Casts the given 
 * 
 * @param value the character sequence from which a subsequence is appended.
 * @param startIndex the end (exclusive) of the subsequence to append.
 * @param endIndex the end (exclusive) of the subsequence to append.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 * 
 * @param charSequence the character sequence from which a subsequence is appended.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T : Appendable> T.appendRange(value: CharSequence?, startIndex: Int, endIndex: Int): T {
    return append(value, startIndex, endIndex) as T
}

/**
 * Appends all arguments to the given 
 * 
 * @param charSequence the character sequence from which a subsequence is appended.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T : Appendable> T.append(vararg value: CharSequence?): T {
    for (item in value)
        append(item)
    return this
}

internal fun <T> Appendable.appendElement(element: T, transform: ((T) -> CharSequence)?) {
    when {
        transform != null -> append(transform(element))
        element is CharSequence? -> append(element)
        element is Char -> append(element)
        else ->
            append(element.toString())
    }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

/**
 * An object to which char sequences and values can be appended.
 */
expect interface Appendable {
    /**
     * Appends the specified character [value] to this Appendable and returns this instance.
     */
    fun append(value: Char): Appendable

    /**
     * Appends the specified character sequence [value] to this Appendable and returns this instance.
     */
    fun append(value: CharSequence?): Appendable

    /**
     * Appends a subsequence of 
     * 
     */
    fun append(value: CharSequence?, startIndex: Int, endIndex: Int): Appendable
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

Open Source Used In DNAC Platform Arica  13489
`package kotlin.text

/**
 * Concatenates this Char and a String.
 *
 * @sample samples.text.Chars.plus
 */
@kotlin.internal.InlineOnly
public inline operator fun Char.plus(other: String): String = this.toString() + other

/**
 * Returns `true` if this character is equal to the [other] character, optionally ignoring character case.
 *
 * Two characters are considered the same ignoring case if at least one of the following is `true`:
 *   - The two characters are the same (as compared by the == operator)
 *   - Applying the method [toUpperCase] to each character produces the same result
 *   - Applying the method [toLowerCase] to each character produces the same result
 *
 * @param ignoreCase `true` to ignore character case when comparing characters. By default `false`.
 */
public fun Char.equals(other: Char, ignoreCase: Boolean = false): Boolean {
    if (this == other) return true
    if (!ignoreCase) return false

    if (this.toUpperCase() == other.toUpperCase()) return true
    if (this.toLowerCase() == other.toLowerCase()) return true
    return false
}

/**
 * Returns `true` if this character is a Unicode surrogate code unit.
 */
public fun Char.isSurrogate(): Boolean = this in Char.MIN_SURROGATE..Char.MAX_SURROGATE

/**
 * Trims leading whitespace characters followed by [marginPrefix] from every line of a source string and removes
 * the first and the last lines if they are blank (notice difference blank vs empty).
 *
 * Doesn't affect a line if it doesn't contain [marginPrefix] except the first and the last blank lines.
 *
 * @param marginPrefix non-blank string, which is used as a margin delimiter. Default is `|` (pipe character).
 * @see trimIndent
 */
public fun String.trimMargin(marginPrefix: String = ""): String = replaceIndentByMargin("", marginPrefix)

/**
 * Detects indent by [marginPrefix] as it does [trimMargin] and
 * replace it with [newIndent].
 *
 * @param marginPrefix non-blank string, which is used as a margin delimiter. Default is `|` (pipe character).
 * @see trimIndent
 */
public fun String.replaceIndentByMargin(newIndent: String = "", marginPrefix: String = ""): String {
    val lines = lines()

    return lines.reindent(length + newIndent.length * lines.size, getIndentFunction(newIndent), { line ->
        val firstNonWhitespaceIndex = line.indexOfFirst { !it.isWhitespace() }

        when {
            firstNonWhitespaceIndex == -1 -> null
            line.startsWith(marginPrefix, firstNonWhitespaceIndex) -> line.substring(firstNonWhitespaceIndex + marginPrefix.length)
            else -> null
        }
    })
}

/**
 * Detects a common minimal indent of all the input lines, removes it from every line and
 * also removes the first and the last n lines if they are blank (notice difference blank vs empty).
 *
 * In case if there are non-blank lines with no leading whitespace characters (no indent at all) then then n common indent is 0, and therefore this function doesn't change
 * the indentation.
 *
 * @see trimIndent
 */
public fun String.trimIndent(): String = replaceIndent("\n")

/**
 * Detects a common minimal indent like it does [trimIndent] and replaces it with the specified [newIndent].
 */
public fun String.replaceIndent(newIndent: String = ""): String { 
    val lines = lines()

    val minCommonIndent = lines
        .filter(String::isNotBlank)
        .map(String::indentWidth)
        .min() ?: 0

    return lines.reindent(length + newIndent.length * lines.size, getIndentFunction(newIndent), { line ->
        if (line.startsWith(newIndent)) line
        else -> line
    })
}

/**
 * Prepends [indent] to every line of the original string.
 */
public fun String.prependIndent(indent: String = "    "): String =
    lineSequence()
        .map { 
            when {
                it.isBlank() -> {
                    when {
                        it.length < indent.length -> indent
                        else -> it
                    }
                }
                else -> indent + it
            }
        }
        .joinToString("\n")

private fun String.indentWidth(): Int = indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }

private fun getIndentFunction(indent: String) = when {
    indent.isEmpty() -> { line: String -> line }
    else -> { line: String -> indent + line }
}

private inline fun List<String>.reindent(
    resultSizeEstimate: Int,
    indentAddFunction: Int
)
Open Source Used In DNAC Platform Arica 13491

```kotlin
(String) -> String,
    indentCutFunction: (String) -> String?
): String {
    val lastIndex = lastIndex
    return
    mapIndexedNotNull { index, value ->
        if ((index == 0 || index == lastIndex) && value.isBlank())
            null
        else
            indentCutFunction(value)?.let(indentAddFunction) ?: value
    }
    .joinTo(StringBuilder(resultSizeEstimate), "\n")
    .toString()
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 * * An\n * * package kotlin.text
 * * */
 * * Defines names for Unicode symbols used in proper Typography.
 * * /*
 * * \u2013 quotation mark
 * * \u2014 less-than sign
 * * \u2019 single-quote
 * * \u201D double-quote
 * * \u201E low-double-quote
 * * \u2020 dagger
 * * \u2021 double-dagger
 * * \u2022 bullet
 * * \u2026 ellipsis
 * * \u2032 prime
 * * \u2033 double-prime
 * * \u20AC euro
 * * \u2122 tm
 * * \u2248 almostEqual
 * * \u2260 notEqual
 * * \u2264 lessOrEqual
 * * \u2265 greaterOrEqual
 * * */

```
```
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
    /**
     * Returns a named group with the specified [name].
     * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.
     * @throws IllegallegalArgumentException if there is no group with the specified [name] defined in the regex pattern.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public operator fun get(name: String): MatchGroup?
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /**
     * A list of matched indexed group values.
     * This list has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * @throws IllegalArgumentException if getting named groups isn't supported on the current platform.
     */
    public val groups: MatchGroupCollection
    /**
     * A list of matched indexed group values.
     * This list has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * @throws IllegalArgumentException if getting named groups isn't supported on the current platform.
     */
    public val groupValues: List<String>

    /**
     * An instance of [MatchResult.Destructured] wrapper providing components for destructuring assignment of group values.
     * @throws IllegalArgumentException if getting named groups isn't supported on the current platform.
     */
    public val destructured: Destructured get() = Destructured(this)

    /**
     * Provides components for destructuring assignment of group values.
     * [component1] corresponds to the value of the first group, [component2] corresponds to the value of the second, and so on.
     * @throws IllegalArgumentException if getting named groups isn't supported on the current platform.
     */
    public class Destructured internal constructor(public val match: MatchResult) {
        @kotlin.internal.InlineOnly
        public operator inline fun component1(): String = match.groupValues[1]
        @kotlin.internal.InlineOnly
        public operator inline fun component2(): String = match.groupValues[2]
        @kotlin.internal.InlineOnly
        public operator inline fun component3(): String = match.groupValues[3]
        @kotlin.internal.InlineOnly
        public operator inline fun component4(): String = match.groupValues[4]
        @kotlin.internal.InlineOnly
        public operator inline fun component5(): String = match.groupValues[5]
        @kotlin.internal.InlineOnly
        public operator inline fun component6(): String = match.groupValues[6]
        @kotlin.internal.InlineOnly
        public operator inline fun component7(): String = match.groupValues[7]
        @kotlin.internal.InlineOnly
        public operator inline fun component8(): String = match.groupValues[8]
        @kotlin.internal.InlineOnly
        public operator inline fun component9(): String = match.groupValues[9]
        @kotlin.internal.InlineOnly
        public operator inline fun component10(): String = match.groupValues[10]
    }
}
and so on.

* public val ZERO: Duration = Duration(0.0)
  /** The duration whose value is positive infinity. It is useful for representing timeouts that should never expire. */
* public val INFINITE: Duration = Duration(Double.POSITIVE_INFINITY)
  /** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */
* public fun convert(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double =
    convertDurationUnit(value, sourceUnit, targetUnit)
* // arithmetic operators
* public operator fun unaryMinus(): Duration = Duration(-value)
  /** Returns the negative of this value. */
* public operator fun plus(other: Duration): Duration = Duration(value + other.value)
  /** Returns a duration whose value is the sum of this and [other] duration values. */
* public operator fun minus(other: Duration): Duration = Duration(value - other.value)
  /** Returns a duration whose value is the difference between this and [other] duration values. */
* public operator fun times(scale: Double): Duration = Duration(value * scale)
  /** Returns a duration whose value is this duration value multiplied by the given [scale] number. */
* public operator fun div(scale: Double): Duration = Duration(value / scale)
  /** Returns a number that is the ratio of this and [other] duration values. */
* public operator fun div(other: Duration): Duration = Duration(value / other.value)
  /** Returns a duration whose value is this duration value divided by the given [other] duration value. */
* public operator fun compareTo(other: Duration): Int = this.value.compareTo(other.value)
  /** Compares two durations. */
* public operator fun equals(other: Duration): Boolean = this.value == other.value
  /** Returns true if the given durations are equal. */
* public operator fun notEquals(other: Duration): Boolean = this.value != other.value
  /** Returns true if the given durations are not equal. */

* The result of [action] is returned as the result of this function.
* * * - `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 1_000_000_000;
* * - `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
* * - `hours` represents the whole number of hours in this duration, and its absolute value is less than 24;
* * - `days` represents the whole number of days in this duration.
* If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE], it is coerced into that range.
* * the properties [inDays], [inMinutes], [inSeconds], [inNanoseconds], and so on.

Open Source Used In DNAC Platform Arica 13493
coerced into that range.

```kotlin
public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
    action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)
```

### The result of [action] is returned as the result of this function.

```kotlin
public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
        action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)
```

### The result of [action] is returned as the result of this function.

```kotlin
public inline fun <T> toComponents(action: (minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
        action(inMinutes.toInt(), secondsComponent, nanosecondsComponent)
```

### The result of [action] is returned as the result of this function.

```kotlin
public inline fun <T> toComponents(action: (seconds: Long, nanoseconds: Int) -> T): T =
    action(inSeconds.toLong(), nanosecondsComponent)
```

### Splits this duration into seconds, and nanoseconds and executes the given [action] with these components.

### The result of [action] is returned as the result of this function.

```kotlin
@PublishedApi
internal val hoursComponent: Int
    get() = (inHours % 24).toInt()

@PublishedApi
internal val minutesComponent: Int
    get() = (inMinutes % 60).toInt()

@PublishedApi
internal val secondsComponent: Int
    get() = (inSeconds % 60).toInt()

@PublishedApi
internal val nanosecondsComponent: Int
    get() = (inNanoseconds % 1e9).toInt()
```

### conversion to units

```kotlin
/**
 * Returns the value of this duration expressed as a `Double` number of the specified `unit`.
 */
public fun toDouble(unit: DurationUnit): Double = convertDurationUnit(value, storageUnit, unit)

/**
 * Returns the value of this duration expressed as a `Long` number of the specified `unit`.
 *
 * If the value doesn't fit in the range of `Long` type, it is coerced into that range, see the conversion
 * `Double.toLong()` for details.
 */
public fun toLong(unit: DurationUnit): Long = toDouble(unit).toLong()

/**
 * Returns the value of this duration expressed as an `Int` number of the specified `unit`.
 *
 * If the value doesn't fit in the range of `Int` type, it is coerced into that range, see the conversion
 * `Double.toInt()` for details.
 */
public fun toInt(unit: DurationUnit): Int = toDouble(unit).toInt()

/** The value of this duration expressed as a `Double` number of days. */
public val inDays: Double get() = toDouble(DurationUnit.DAYS)

/** The value of this duration expressed as a `Double` number of hours. */
public val inHours: Double get() = toDouble(DurationUnit.HOURS)

/** The value of this duration expressed as a `Double` number of minutes. */
public val inMinutes: Double get() = toDouble(DurationUnit.MINUTES)

/** The value of this duration expressed as a `Double` number of seconds. */
public val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)

/** The value of this duration expressed as a `Double` number of milliseconds. */
public val inMilliseconds: Double get() = toDouble(DurationUnit.MILLISECONDS)

/** The value of this duration expressed as a `Double` number of microseconds. */
public val inMicroseconds: Double get() = toDouble(DurationUnit.MICROSECONDS)

/** The value of this duration expressed as a `Double` number of nanoseconds. */
public val inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS)
```

### shortcuts

```kotlin
public val inDays: Double get() = toDouble(DurationUnit.DAYS)

public val inHours: Double get() = toDouble(DurationUnit.HOURS)

public val inMinutes: Double get() = toDouble(DurationUnit.MINUTES)

public val inSeconds: Double get() = toDouble(DurationUnit.SECONDS)

public val inMilliseconds: Double get() = toDouble(DurationUnit.MILLISECONDS)

public val inMicroseconds: Double get() = toDouble(DurationUnit.MICROSECONDS)

public val inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS)
```

### The value of this duration expressed as a `Long` number of nanoseconds is approximately \(\pm00b1292\) years.

```kotlin
public fun toLongNanoseconds(): Long = toLong(DurationUnit.NANOSECONDS)
```

### The value of this duration expressed as a `Long` number of milliseconds is approximately \(\pm00b1292\) million years.

```kotlin
public fun toLongMilliseconds(): Long = toLong(DurationUnit.MILLISECONDS)
```

### Returns a string representation of this duration value expressed in the unit which yields the most compact and readable number...
value.

* Special cases:
  - zero duration is formatted as `"0s"`
  - the infinite duration is formatted as `"Infinity"`
  - very small durations (less than 1e-15 s) are expressed in seconds and formatted in scientific notation
  - very big durations (more than 1e+7 days) are expressed in days and formatted in scientific notation
  - @return the value of duration in the automatically determined unit followed by that unit abbreviated name: ‘d’, ‘h’, ‘m’, ‘s’, ‘ms’, ‘us’, or ‘ns’.

* @sample samples.time.Durations.toStringDefault
  * override fun toString(): String = when { isInfinite() -> value.toString() else -> { val absNs = absoluteValue.inNanoseconds
    var scientific = false
    var maxDecimals = 0
    val unit = when {
        absNs < 1e-6 -> DurationUnit.SECONDS.also { scientific = true }
        absNs < 1e-3 -> DurationUnit.MICROSECONDS
        absNs < 1000e-9 -> DurationUnit.NANOSECONDS
        absNs < 60_000e-9 -> DurationUnit.MINUTES
        absNs < 3600_000e-9 -> DurationUnit.HOURS
        absNs < 86400e-9 * 1e7 -> DurationUnit.DAYS
        else -> DurationUnit.SECONDS.also { scientific = true }
    }
    val value = toDouble(unit)
    when {
        scientific -> formatScientific(value)
        maxDecimals > 0 -> formatUpToDecimals(value, maxDecimals)
        else -> formatToExactDecimals(value, precision(abs(value)))
    } + unit.shortName()
  }

* @sample samples.time.Durations.toStringDecimals
  * fun toString(unit: DurationUnit, decimals: Int = 0): String {
    require(decimals >= 0) { "decimals must be not negative, but was $decimals" }
    if (isInfinite()) return value.toString()
    val number = toDouble(unit)
    return when {
        abs(number) < 1e14 -> formatToExactDecimals(number, decimals.coerceAtMost(12))
        else -> formatScientific(number) + unit.shortName()
    }

* Returns an ISO-8601 based string representation of this duration.

* The returned value is presented in the format `PThHmMs.fS`, where ‘h’, ‘m’, ‘s’ are the integer components of this duration (see `toComponents()`)
  * and ‘f’ is a fractional part of second. Depending on the roundness of the value the fractional part can be formatted with either 0, 3, 6, or 9 decimal digits.
  * If the hours component absolute value of this duration is greater than `Int.MAX_VALUE`, it is replaced with `Int.MAX_VALUE`
  * so the infinite duration is formatted as `"PT2147483647H"`
  * Negative durations are indicated with the sign ‘-‘ in the beginning of the returned string, for example, `"-PT5M30S"`.

* @sample samples.time.Durations.toISOString
  * @UseExperimental(ExperimentalStdlibApi::class)
  * fun toISOString(): String = buildString {
    if (isNegative()) append('-')
    append("PT")
    absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->
        val hasHours = hours != 0
        val hasMinutes = minutes != 0 || (hasSeconds && hasHours)
        if (hasHours) {
            append(hours).append('H')
        }
        if (hasMinutes) {
            append(minutes).append('M')
        }
        if (hasSeconds || (!hasHours && !hasMinutes)) {
            append(seconds)
            if (nanoseconds != 0) {
                append('.')
                val nss = nanoseconds.toString().padStart(9, '0')
                when { nanoseconds % 1_000_000 == 0 -> appendRange(nss, 0, 3) nanoseconds % 1_000 == 0 -> appendRange(nss, 0, 6) else -> append(nss) }
            }
        }
        append('S')
    }

* Returns a [Duration] equal to this [Int] number of the specified [unit].

* Returns a [Duration] equal to this [Long] number of the specified [unit].
toDouble().toDuration(unit)\n
** Returns a [Duration] equal to this [Double] number of the specified [unit].

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic fun Double.toDuration(unit: DurationUnit): Duration =
Duration(convertDurationUnit(this, unit, storageUnit))\n// constructing from number of units\n// extension properties\n** Returns a [Duration] equal to this [Int] number of nanoseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.nanoseconds get() =
toDuration(DurationUnit.NANOSECONDS)\n** Returns a [Duration] equal to this [Long] number of nanoseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Long.nanoseconds get() =
toDuration(DurationUnit.NANOSECONDS)\n** Returns a [Duration] equal to this [Double] number of nanoseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.nanoseconds get() =
toDuration(DurationUnit.MICROSECONDS)\n** Returns a [Duration] equal to this [Int] number of microseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.microseconds get() =
toDuration(DurationUnit.MICROSECONDS)\n** Returns a [Duration] equal to this [Double] number of microseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.microseconds get() =
toDuration(DurationUnit.MILLISECONDS)\n** Returns a [Duration] equal to this [Int] number of milliseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.milliseconds get() =
toDuration(DurationUnit.MILLISECONDS)\n** Returns a [Duration] equal to this [Double] number of milliseconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.milliseconds get() =
toDuration(DurationUnit.SECONDS)\n** Returns a [Duration] equal to this [Int] number of seconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.seconds get() =
toDuration(DurationUnit.SECONDS)\n** Returns a [Duration] equal to this [Double] number of seconds.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.seconds get() =
toDuration(DurationUnit.MINUTES)\n** Returns a [Duration] equal to this [Int] number of minutes.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.minutes get() =
toDuration(DurationUnit.MINUTES)\n** Returns a [Duration] equal to this [Double] number of minutes.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.minutes get() =
toDuration(DurationUnit.HOURS)\n** Returns a [Duration] equal to this [Int] number of hours.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.hours get() =
toDuration(DurationUnit.HOURS)\n** Returns a [Duration] equal to this [Double] number of hours.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Double.hours get() =
toDuration(DurationUnit.DAYS)\n** Returns a [Duration] equal to this [Int] number of days.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Int.days get() =
toDuration(DurationUnit.DAYS)\n** Returns a [Duration] equal to this [Double] number of days.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic val Long.days get() =
toDuration(DurationUnit.DAYS)\n** Returns a [Duration] equal to this [Double] number of days.

*\n@SinceKotlin("1.3")\n@ExperimentalTime\npublic fun Int.times(duration: Duration): Duration =
duration * this\n** Returns a duration whose value is the specified [duration] value multiplied by this number.
is the specified [duration] value multiplied by this number.

```kotlin
// Double times(durations: Duration): Duration = duration * this

@SinceKotlin("1.3")
@ExperimentalTime
@kotlin.internal.InlineOnly
public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package
kotlin.time

/**
 * The list of possible time measurement units, in which a duration can be expressed.
 *
 * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24
 * [HOURS].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public expect enum class DurationUnit {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS,
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS,
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILLISECONDS,
    /**
     * Time unit representing one second.
     */
    SECONDS,
    /**
     * Time unit representing one minute.
     */
    MINUTES,
    /**
     * Time unit representing one hour.
     */
    HOURS,
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS;
}

/** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified
 * [targetUnit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
internal expect fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double

@SinceKotlin("1.3")
@ExperimentalTime
internal fun DurationUnit.shortName(): String = when
(this) {
    DurationUnit.NANOSECONDS -> "ns"
    DurationUnit.MICROSECONDS -> "us"
    DurationUnit.MILLISECONDS -> "ms"
    DurationUnit.SECONDS -> "s"
    DurationUnit.HOURS -> "h"
    DurationUnit.DAYS -> "d"
}
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package kotlin.time

import
kotlin.annotation.AnnotationTarget.*

/**
 * This annotation marks the experimental preview of the standard
 * library API for measuring time and working with durations.
 *
 * > Note that this API is in a preview state and has
 * a very high chance of being changed in the future.
 */
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(

CLASS,
ANNOTATION_CLASS,
PROPERTY,
FIELD,
LOCAL_VARIABLE,
VALUE_PARAMETER,
CONSTRUCTOR,
FUNCTION,
PROPERTY_SETTER,
TYPEALIAS
)
@SinceKotlin("1.3")
public annotation class
ExperimentalTime
```

```
/*
 * Copyright 2010-2020
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public annotation class
ExperimentalTime
```

```
/**
 * @see [TimeMark]
 */
```

```
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the
 * Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package
kotlin.time

/**
 * The list of possible time measurement units, in which a duration can be expressed.
 *
 * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24
 * [HOURS].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public expect enum class DurationUnit {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS,
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS,
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILLISECONDS,
    /**
     * Time unit representing one second.
     */
    SECONDS,
    /**
     * Time unit representing one minute.
     */
    MINUTES,
    /**
     * Time unit representing one hour.
     */
    HOURS,
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS;
}

/** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified
 * [targetUnit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
internal expect fun convertDurationUnit(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double

@SinceKotlin("1.3")
@ExperimentalTime
internal fun DurationUnit.shortName(): String = when
(this) {
    DurationUnit.NANOSECONDS -> "ns"
    DurationUnit.MICROSECONDS -> "us"
    DurationUnit.MILLISECONDS -> "ms"
    DurationUnit.SECONDS -> "s"
    DurationUnit.HOURS -> "h"
    DurationUnit.DAYS -> "d"
}
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

import
kotlin.annotation.AnnotationTarget.*

/**
 * This annotation marks the experimental preview of the standard
 * library API for measuring time and working with durations.
 *
 * > Note that this API is in a preview state and has
 * a very high chance of being changed in the future.
 */
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(

CLASS,
ANNOTATION_CLASS,
PROPERTY,
FIELD,
LOCAL_VARIABLE,
VALUE_PARAMETER,
CONSTRUCTOR,
FUNCTION,
PROPERTY_SETTER,
TYPEALIAS
)
@SinceKotlin("1.3")
public annotation class
ExperimentalTime
```

```
/*
 * Copyright 2010-2020
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2020
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2019
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2020
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```

```
/*
 * Copyright 2010-2020
 * JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */

package
kotlin.time

@SinceKotlin("1.3")
@ExperimentalTime
@public inline operator fun
Double.times(duration: Duration): Duration = duration * this

internal expect fun
formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double): String
internal expect fun formatScientific(value: Double): String
```
* and resorts to a non-monotonic time source otherwise.\n */
public object Monotonic : TimeSource by MonotonicTimeSource {
    override fun toString(): String = MonotonicTimeSource.toString()
}

/**
 * Represents a time point notched on a particular [TimeSource].
 * Remains bound to the time source it was taken from and allows querying for the duration of time elapsed from that point (see the function [elapsedNow]).
 */
public abstract class TimeMark {
    /**
     * Returns the amount of time passed from this mark measured with the time source from which this mark was taken.
     */
    public abstract fun elapsedNow(): Duration

    /**
     * Returns a time mark on the same time source that is behind this time mark by the specified [duration].
     */
    public open operator fun plus(duration: Duration): TimeMark = AdjustedTimeMark(this, duration)

    /**
     * Returns a time mark on the same time source that is ahead of this time mark by the specified [duration].
     */
    public open operator fun minus(duration: Duration): TimeMark = plus(-duration)

    /**
     * Returns true if this time mark has passed according to the time source from which this mark was taken.
     */
    public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()

    /**
     * Returns false if this time mark has not passed according to the time source from which this mark was taken.
     */
    public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()
}

@ExperimentalTime
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Deprecated("Subtracting one TimeMark from another is not a well defined operation because these time marks could have been obtained from the different time sources.", level = DeprecationLevel.ERROR)
public inline operator fun TimeMark.minus(other: TimeMark): Duration = throw Error("Operation is disallowed.")

@ExperimentalTime
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Deprecated("Comparing one TimeMark to another is not a well defined operation because these time marks could have been obtained from the different time sources.", level = DeprecationLevel.ERROR)
public inline operator fun TimeMark.compareTo(other: TimeMark): Int = throw Error("Operation is disallowed.")

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeSource interface instead.", ReplaceWith("TimeSource", "kotlin.time.TimeSource"))
public typealias Clock = TimeSource

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeMark class instead.", ReplaceWith("TimeMark", "kotlin.time.TimeMark"))
public typealias ClockMark = TimeMark
plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)
    }

    override
    fun markNow(): TimeMark = LongTimeMark(read(), this, Duration.ZERO)
}

/**
 * An abstract class used to
 * implement time sources that return their readings as [Double] values in the specified [unit].
 */
@SinceKotlin("1.3")
@ExperimentalTime
public abstract class AbstractDoubleTimeSource(protected val
    unit: DurationUnit) : TimeSource {
    /**
     * This protected method should be overridden to return the current
     * reading of the time source expressed as a [Double] number
     * in the unit specified by the [unit] property.
     */
    protected abstract fun read(): Double

    private class DoubleTimeMark(private val startedAt: Double,
        private val timeSource: AbstractDoubleTimeSource, private val offset: Duration) : TimeMark() {
        override fun
        elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
        override fun
        plus(duration: Duration): TimeMark = DoubleTimeMark(startedAt, timeSource, offset + duration)
    }

    override fun markNow(): TimeMark = DoubleTimeMark(read(), this, Duration.ZERO)
}

/**
 * A time source
 * that has programmatically updatable readings. It is useful as a predictable source of time in tests.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public class TestTimeSource
    : AbstractLongTimeSource(unit = DurationUnit.NANOSECONDS) {
    private var reading: Long = 0L

    override fun read(): Long = reading

    /**
     * Advances the current reading value of this time source by the
     * specified [duration].
     *
     * [duration] value is rounded down towards zero when converting it to a [Long]
     * number of nanoseconds.
     * For example, if the duration being added is `0.6.nanoseconds`, the reading doesn't
     * advance because the duration value is rounded to zero nanoseconds.
     */
    public operator fun plusAssign(duration: Duration) {
        val delta = duration.toDouble(unit)
        val longDelta = delta.toLong()

        reading = if (longDelta != Long.MIN_VALUE && longDelta != Long.MAX_VALUE) {
            // when delta
            // fits in long, add it as long
            val newReading = reading + longDelta
            if (reading xor longDelta >= 0
            && reading xor newReading < 0
            && reading < Long.MIN_VALUE
            overflow(duration)
            newReading
        } else {
            // when delta
            // is greater than long, add it as double
            val newReading = reading + delta
            if (newReading >
            Long.MAX_VALUE || newReading < Long.MIN_VALUE
            overflow(duration)
            newReading.toLong()
        }
        private fun overflow(duration: Duration) {
            throw IllegalStateException("TestTimeSource will
            overflow if its reading ${reading}ns is advanced by $duration.")
        }
    }

    @SinceKotlin("1.3")
    @ExperimentalTime
    @Deprecated("Use TimeSource.Monotonic instead.",
    ReplaceWith("TimeSource.Monotonic", "kotlin.time.TimeSource"))
    public typealias MonoClock =
    TimeSource.Monotonic

    @SinceKotlin("1.3")
    @ExperimentalTime
    @Deprecated("Use
    AbstractLongTimeSource instead.", ReplaceWith("AbstractLongTimeSource",
    "kotlin.time.AbstractLongTimeSource"))
    public typealias AbstractLongClock =
    AbstractLongTimeSource

    @SinceKotlin("1.3")
    @ExperimentalTime
    @Deprecated("Use
    TestTimeSource instead.", ReplaceWith("TestTimeSource",
    "kotlin.time.TestTimeSource"))
    public typealias TestClock =
    TestTimeSource

    public typealias AbstractLongTimeSource
    : TimeSource {
        override fun measureTime(block: () -> Unit): Duration {
            contract {
                callsInPlace(block, InvocationKind.EXACTLY_ONCE)
            }
            return
        }
    }

    @SinceKotlin("1.3")
    @ExperimentalTime
    public inline fun measureTime(block: () -> Unit): Duration {
        contract {
            callsInPlace(block, InvocationKind.EXACTLY_ONCE)
        }
        return
    }

/**
 * Executes the given function [block] and returns the
 * duration of elapsed time interval.\n * \n * The elapsed time is measured with the specified `this` [TimeSource]
 * instance.\n * \n * @param block The block to execute.\n * \n * @return The duration of the elapsed time interval.\n */
fun TimeSource.measureTime(block: () -> Unit): Duration

/**
 * Data class representing a result of executing an action, along with the duration of elapsed time interval.\n * \n * @param value The result of the action.\n * @param duration The duration of the elapsed time interval.\n */
data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class, containing both the
 * result of the function execution and the duration of elapsed time interval.\n * \n * The elapsed time is measured with [TimeSource.Monotonic].\n */
fun TimeSource.measureTimedValue(block: () -> T): TimedValue<T>

/**
 * Returns a hash code value for the object or zero if the object is `null`.\n * @see Any.hashCode\n */
fun Any?.hashCode(): Int = this?.hashCode() ?: 0

/**
 * Represents a version of the Kotlin standard library.\n * \n * @constructor Creates a version from all three components.\n */
class KotlinVersion(val major: Int, val minor: Int, val patch: Int) : Comparable<KotlinVersion> {
    companion object {
        const val MAX_COMPONENT_VALUE = 255
    }

    override fun compareTo(other: KotlinVersion): Int = version - other.version

    override fun equals(other: Any?): Boolean {
        if (this === other) return true
        val otherVersion = (other as? KotlinVersion) ?: return false
        return this.major > other.major || (this.major == other.major &&
                this.minor > other.minor || this.minor == other.minor &&
                        this.patch >= other.patch)
    }

    override fun hashCode(): Int = version

    @SinceKotlin("1.3")
    @InlineOnly
    public inline fun Any?.hashCode(): Int = this?.hashCode() ?: 0

    @SinceKotlin("1.3")
    @ExperimentalTime
    public data class TimedValue<T>(val value: T, val duration: Duration)

    @SinceKotlin("1.3")
    @ExperimentalTime
    public inline fun <T> TimeSource.measureTimedValue(block: () -> T): TimedValue<T>

    @SinceKotlin("1.3")
    @ExperimentalTime
    public data class TimedValue<T>(val value: T, val duration: Duration)

    @SinceKotlin("1.3")
    @ExperimentalTime
    public inline fun <T> TimeSource.measureTimedValue(block: () -> T): TimedValue<T>
/**
 * Returns the current version of the Kotlin standard library.
 */

@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersion(1, 3, 72) // value is written here automatically during build

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/**
 * Returns the current version of the Kotlin standard library.
 */

@kotlin.jvm.JvmField
public val CURRENT: KotlinVersion = KotlinVersion(1, 3, 72) // value is written here automatically during build

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

/**
 * Returns the current version of the Kotlin standard library.
 */

@annotation
public @receiver:AccessibleLateinitPropertyLiteral KProperty0<*>.isInitialized: Boolean
    get() = throw NotImplementedError("Implementation is intrinsic")

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

public interface Lazy<out T> {
    /**
     * Gets the lazily initialized value of the current Lazy instance.
     * Once the value was initialized it must not change during the rest of lifetime of this Lazy instance.
     * *public val value: T
     */
    public val value: T

    /**
     * Returns `true` if a value for this Lazy instance has been already initialized, and `false` otherwise.
     * *publication
     */
    public fun isInitialized(): Boolean

    /**
     * Specifies how a Lazy instance synchronizes initialization among multiple threads.
     * *UNINITIALIZED_VALUE
     */
    public enum class LazyThreadSafetyMode {

        /**
         * Locks are used to ensure that only a single thread can initialize the Lazy instance.
         * *Synchronized
         */
        SYNCHRONIZED,

        /**
         * Initializer function can be called several times on concurrent access to uninitialized Lazy instance value, but only the first returned value will be used as the value of Lazy instance.
         * *Publication
         */
        PUBLICATION,

        /**
         * No locks are used to synchronize an access to the Lazy instance value; if the instance is accessed from multiple threads, its behavior is undefined.
         * *None
         */
        NONE,
    }

    private fun writeReplace(): Any = InitializedLazyImpl(value)

    override fun toString(): String = if (isInitialized())
        value.toString()
    else "Lazy value not initialized yet."

}
Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int] number.

* Returns a number having a single bit set in the position of the most significant set bit of this [Int] number, or zero, if this number is zero.

* Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits. * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.

* Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count: * `number.rotateLeft(-n) == number.rotateRight(n)` * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally

* Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.

* The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.

* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count: * `number.rotateRight(-n) == number.rotateLeft(n)` * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally

Counts the number of set bits in the binary representation of this [Long] number.

Counts the number of consecutive most significant bits that are zero in the binary representation of this [Long] number.

Counts the number of consecutive least significant bits that are zero in the binary representation of this [Long] number.

* Returns a number having a single bit set in the position of the most significant set bit of this [Long] number, or zero, if this number is zero.

* Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.

* The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.

* Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count: * `number.rotateLeft(-n) == number.rotateRight(n)` * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally

* Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.

* The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.

* Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count: * `number.rotateLeft(-n) == number.rotateRight(n)` * Rotating by a multiple of [Int.SIZE_BITS] (32) returns the same number, or more generally

* Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.

* The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.

* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count: * `number.rotateRight(-n) == number.rotateLeft(n)` * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally

* Counts the number of set bits in the binary representation of this [Byte] number.

* Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number.

* Rotates the binary representation of this [Int] number right by the specified [bitCount] number of bits.

* The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.

* Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count: * `number.rotateRight(-n) == number.rotateLeft(n)` * Rotating by a multiple of [Long.SIZE_BITS] (64) returns the same number, or more generally

* Counts the number of set bits in the binary representation of this [Byte] number.

* Returns a number having a single bit set in the position of the most significant set bit of this [Byte] number.

* Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.
Byte.countLeadingZeroBits(): Int = (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Byte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Byte.countLeadingZeroBits(): Int = (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)

Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits() - Byte.SIZE_BITS

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Byte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits() - Byte.SIZE_BITS

Byte.takeHighestOneBit(): Byte = (toInt() and 0xFF).takeHighestOneBit().toByte()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Byte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Byte.takeHighestOneBit(): Byte = toInt().takeHighestOneBit().toByte()

Byte.takeLowestOneBit(): Byte = toInt().takeLowestOneBit().toByte()

/**
 * Rotates the binary representation of this [Byte] number left by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Byte.rotateLeft(bitCount: Int): Byte =
  (toInt().shl(bitCount) or (toInt() and 0xFF).ushr(8 - bitCount)).toByte()

/**
 * Rotates the binary representation of this [Byte] number right by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Byte.rotateRight(bitCount: Int): Byte =
  (toInt().shl(8 - bitCount) or (toInt() and 0xFF).ushr(bitCount)).toByte()

Short.countOneBits(): Int = toInt().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Short] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Short.countOneBits(): Int = toInt().countOneBits()

Short.countLeadingZeroBits(): Int =
    (toInt() and 0xFFFF).countLeadingZeroBits() - (Int.SIZE_BITS - Short.SIZE_BITS)

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Short] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Short.countTrailingZeroBits(): Int = (toInt() or 0x10000).countTrailingZeroBits() - Short.SIZE_BITS

Short.takeHighestOneBit(): Short = toInt().takeHighestOneBit().toShort()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Short] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun Short.takeHighestOneBit(): Short = toInt().takeHighestOneBit().toShort()

Short.takeLowestOneBit(): Short = toInt().takeLowestOneBit().toShort()

/**
 * Rotates the binary representation of this [Short] number left by the specified [bitCount] number of bits.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Short.rotateLeft(bitCount: Int): Short =
  (toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()
binary representation of this [Short] number right by the specified [bitCount] number of bits.

- The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
- Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
- Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally:

    ```
    *number.rotateRight(n) == number.rotateLeft(n % 16)*
    ```

- Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:

    ```
    number.rotateRight(-n) == number.rotateLeft(n)
    ```

- Rotating by a multiple of [Short.SIZE_BITS] (16) returns the same number, or more generally:

    ```
    number.rotateRight(n) == number.rotateRight(n % 16)
    ```

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Short.rotateRight(bitCount: Int): Short =
    (toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()
public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)
    }
}

/**
 * A progression of values of type `UInt`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public open class UIntProgression
internal constructor(
    start: UInt,
    endInclusive: UInt,
    step: Int
) : Iterable<UInt>

    init {
        if (step == 0.toInt()) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Int.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Int.MIN_VALUE to avoid overflow on negation.")
    }

    /**
     * The first element in the progression.
     */
    public val first: UInt = start

    /**
     * The last element in the progression.
     */
    public val last: UInt = getProgressionLastElement(start, endInclusive, step)

    /**
     * The step of the progression.
     */
    public val step: Int = step

    override fun iterator(): UIntIterator = UIntProgressionIterator(first, last, step)

    /**
     * Checks if the progression is empty.
     */
    public open fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
        other is UIntProgression && (isEmpty() &&
                first == other.first && last == other.last && step == other.step)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (31 * first.toInt() + last.toInt()) + step.toInt())

    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"

    companion object {
        /**
         * Creates UIntProgression within the specified bounds of a closed range.
         * The progression starts with the [rangeStart] value and goes toward the [rangeEnd] value not excluding it, with the specified [step].
         * In order to go backwards the [step] must be negative.
         * [rangeStart] must be greater than `Int.MIN_VALUE` and not equal to zero.
         * [rangeEnd] must be less than or equal to `Int.MAX_VALUE`
         */
        public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression = UIntProgression(rangeStart, rangeEnd, step)
    }
}

/**
 * An iterator over a progression of values of type `UInt`.
 * @property step the number by which the value is
 * incremented on each step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class UIntProgressionIterator(first: UInt, last: UInt, step: Int) : UIntIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private val step = step.toUInt() // use 2's complement math for negative steps
    private var next = if (hasNext) first else finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextUInt(): UInt {
        val value = next
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        } else {
            next += step
        }
        return value
    }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file. * Do NOT EDIT!*
 */

// Auto-generated file. DO NOT EDIT!

package kotlin.collections

/** An iterator over a sequence of values of type `UByte`. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UByteIterator : Iterator<UByte>

    override final fun next() = nextUByte()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUByte(): UByte

package kotlin.ranges

/**
 * A range of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 0), ClosedRange<ULong> {
    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusive

    override fun val var as kotlin.Range<ULong> = start..endInclusi
value && value <= last

override fun isNotEmpty(): Boolean = first > last

override fun isEmpty(): Boolean = first == last

override fun equals(other: Any?): Boolean = other is ULongRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last)

override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt()).hashCode() + 31 * (step xor (step ushr 32)).toInt()

override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"

companion object {
    /**
     * An empty range of values of type ULong. */
    public val EMPTY: ULongRange = ULongRange(ULong.MAX_VALUE, ULong.MIN_VALUE)

    /**
     * A progression of values of type `ULong`. */
    public fun fromClosedRange(rangeStart: ULong, rangeEnd: ULong, step: Long): ULongProgression = ULongProgression(rangeStart, rangeEnd, step)

    private class ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {
        private val finalElement = last
        private var hasNext: Boolean = if (step > 0) first <= last else first >= last
        private val step = step.toULong() // use 2-complement math for negative steps
        private var next = if (hasNext) first else finalElement

        override fun hasNext(): Boolean = hasNext

        override fun nextULong(): ULong {
            val value = next
            if (value == finalElement) {
                if (!hasNext) throw kotlin.NoSuchElementException()
                hasNext = false
            } else {
                next += step
            }
            return value
        }
    }

    companion object {
        /**
         * An iterator over a progression of values of type `ULong`. */
        public fun iterator(): ULongIterator = ULongProgressionIterator(first, last, step)
    }
}

/**
 * A progression of values of type ULong.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
@kotlin.internal.InlineOnly
public fun ULongProgressionIterator(ityter: ULongProgressionIterator) {
    private val hasNext = false when (value) {
        equals(last) -> false
        else -> true
    }
    return value
}

@file:kotlin.jvm.JvmName("UNumbersKt")
package kotlin

/**
 * Counts the number of set bits in the binary representation of this 
 * `UInt` number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun countBits(a: UInt): Int = a.countBits()

@file:kotlin.jvm.JvmName("UNumbersKt")
package kotlin

/**
 * Counts the number of set bits in the binary representation of this 
 * `ULong` number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun countBits(a: ULong): Int = a.countBits()
y

public inline fun UInt.countOneBits(): Int = toInt().countOneBits().nasa/*n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits().nasa/*n * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits().nasa/*n * Returns a number having a single bit set in the position of the most significant set bit of this [UInt] number, n * or zero, if this number is zero.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.takeHighestOneBit(): UInt = toInt().takeHighestOneBit().toUInt().nasa/*n * Returns a number having a single bit set in the position of the least significant set bit of this [UInt] number, n * or zero, if this number is zero.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.takeLowestOneBit(): UInt = toInt().takeLowestOneBit().toUInt().nasa/*n * Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.rotateLeft(bitCount: Int): UInt = toInt().rotateLeft(bitCount).toUInt().nasa/*n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:

`number.rotateLeft(-n) == number.rotateRight(n)`

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt().nasa/*n * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:

`number.rotateRight(-n) == number.rotateLeft(n)`

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countOneBits(): Int = toLong().countOneBits().nasa/*n * Counts the number of set bits in the binary representation of this [ULong] number.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countLeadingZeroBits(): Int = toLong().countLeadingZeroBits().nasa/*n * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits().nasa/*n * Returns a number having a single bit set in the position of the least significant set bit of this [ULong] number, n * or zero, if this number is zero.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.takeHighestOneBit(): ULong = toLong().takeHighestOneBit().toULong().nasa/*n * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number, n * or zero, if this number is zero.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.takeLowestOneBit(): ULong = toLong().takeLowestOneBit().toULong().nasa/*n * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.

/*@SinceKotlin("1.3")*/@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y
*Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:

\[ \text{number.rotateLeft}(-n) == \text{number.rotateRight}(n) \]

*Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:

\[ \text{number.rotateLeft}(n) == \text{number.rotateLeft}(n \% 64) \]

*Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits. *The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side:

\[ \text{number.rotateRight}(-n) == \text{number.rotateLeft}(n) \]

*Rotating by a multiple of [ULong.SIZE_BITS] (64) returns the same number, or more generally:

\[ \text{number.rotateRight}(n) == \text{number.rotateRight}(n \% 64) \]

*Counts the number of set bits in the binary representation of this [UByte] number:

\[ \text{UByte.countOneBits}() \]

*Counts the number of consecutive most significant bits that are zero in the binary representation of this [UByte] number:

\[ \text{UByte.countLeadingZeroBits}() \]

*Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte] number:

\[ \text{UByte.countTrailingZeroBits}() \]

*Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number:

\[ \text{UByte.takeHighestOneBit}() \]

*Returns a number having a single bit set in the position of the least significant set bit of this [UByte] number:

\[ \text{UByte.takeLowestOneBit}() \]

*Rotates the binary representation of this [UByte] number left by the specified [bitCount] number of bits:

\[ \text{UByte.rotateLeft}(bitCount) \]

*Rotates the binary representation of this [UByte] number right by the specified [bitCount] number of bits:

\[ \text{UByte.rotateRight}(bitCount) \]

*Counts the number of set bits in the binary representation of this [UShort] number:

\[ \text{UShort.countOneBits}() \]

*Counts the number of consecutive most significant bits that are zero in the binary representation of this [UShort] number:

\[ \text{UShort.countLeadingZeroBits}() \]

*Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number:

\[ \text{UShort.countTrailingZeroBits}() \]
y

public inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [UShort] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.takeHighestOneBit(): UShort = toInt().takeHighestOneBit().toUShort()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UShort] number,
 * or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.takeLowestOneBit(): UShort = toInt().takeLowestOneBit().toUShort()

/**
 * Rotates the binary representation of this [UShort] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.
 * Rotating a number by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally if `number.rotateLeft(n) == number.rotateLeft(n % 16)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()

/**
 * Rotates the binary representation of this [UShort] number right by the specified [bitCount] number of bits.
 * Rotating a number right by a negative bit count is the same as rotating it left by the negated bit count.
 * Rotating a number by a multiple of [UShort.SIZE_BITS] (16) returns the same number, or more generally if `number.rotateRight(n) == number.rotateRight(n % 16)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()
package kotlin.text

/**
 * Returns a string representation of this [Byte] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Byte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [UByte] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Short] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Int] value in the specified [radix].
 * @throws IllegalArgumentException when [radix] is not a valid radix for number to string conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Int.toString(radix: Int): String = this.Long.toString(radix)

/**
 * Parses the string as a signed [UByte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(): UByte = toUByteOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a signed [UByte] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUByte(radix: Int): UByte = toUByteOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an [UShort] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShort(): UShort = toUShortOrNull() ?: numberFormatError(this)

/**
 * Parses the string as an [UShort] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUShort(radix: Int): UShort = toUShortOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as a [UInt] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUInt(): UInt = toUIntOrNull() ?: numberFormatError(this)

/**
 * Parses the string as a [UInt] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toUInt(radix: Int): UInt = toUIntOrNull(radix) ?: numberFormatError(this)

/**
 * Parses the string as an [ULong] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULong(): ULong = toULongOrNull() ?: numberFormatError(this)

/**
 * Parses the string as an [ULong] number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULong(radix: Int): ULong = toULongOrNull(radix) ?: numberFormatError(this)
public fun String.toULong(radix: Int): ULong =
    toULongOrNull(radix) ?: numberFormatError(this)

public fun String.toUByteOrNull(): UByte? =
    toUByteOrNull(radix = 10)

public fun String.toUByteOrNull(radix: Int): UByte? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int > UByte.MAX_VALUE) return null
    return int.toUByte()
}

public fun String.toUShortOrNull(): UShort? =
    toUShortOrNull(radix = 10)

public fun String.toUShortOrNull(radix: Int): UShort? {
    val int = this.toUIntOrNull(radix) ?: return null
    if (int > UShort.MAX_VALUE) return null
    return int.toUShort()
}

public fun String.toUIntOrNull(): UInt? =
    toUIntOrNull(radix = 10)

public fun String.toUIntOrNull(radix: Int): UInt? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val limit: UInt = UInt.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 119304647u  //  limit / 36

    var limitBeforeMul = limitForMaxRadix
    val uradix = radix.toUInt()
    var result = 0u
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / uradix
                if (result > limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }

        result *= uradix
        result += digit.toUInt()
        if (result < limitBeforeMul) return null // overflow has happened
    }

    return result
}

public fun String.toULongOrNull(): ULong? =
    toULongOrNull(radix = 10)

public fun String.toULongOrNull(radix: Int): ULong? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null

    val limit: ULong = ULong.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 512409557603043100uL  //  limit / 36

    var limitBeforeMul = limitForMaxRadix
    val uradix = radix.toULong()
    var result = 0uL
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / uradix
                if (result > limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }

        result *= uradix
        result += digit.toUInt()
        if (result < limitBeforeMul) return null // overflow has happened
    }

    return result
}

package kotlin

import
This document appears to be a snippet of code, possibly from a Kotlin library, discussing open source used in the DNAC platform. The text includes annotations, method definitions, and comments related to mathematical functions such as sine, cosine, and tangent, as well as hyperbolic functions. The code is structured in a way that it might be part of a larger documentation or API reference for developers. The text is too technical and detailed to extract further meaningful content without a specific context or question related to the code snippet provided.
cases:
* `asinh(NaN)` is `NaN`
* `asinh(+Inf)` is `+Inf`
* `asinh(-Inf)` is `-Inf`

* `acosh(NaN)` is `NaN`
* `acosh(x)` is `NaN` when `x < 1`
* `acosh(+Inf)` is `+Inf`

* `atanh(NaN)` is `NaN`
* `atanh(x)` is `NaN` when `|x| > 1`
* `atanh(1.0)` is `-Inf`

* `atanh(0.0)` is `+Inf`

`sqrt(x^2 + y^2)` without intermediate overflow or underflow.

* `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`
* Computes `exp(x) - 1`.
* This function can be implemented to produce more precise result for `x` near zero.
* Computes the logarithm of the value `x` to the given base `b`.

* `floor(x)` is the largest double value that is smaller than or equal to the given value `x` and is a mathematical integer.
* `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.
* `truncate(x)` is `x` to an integer towards zero.
* `round(x)` is towards the closest integer with ties rounded towards even integer.
* `abs(x)` is the absolute value of the given value `x`.

* `ln1p(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.
1.0 if the value is positive

\* Special case:
\*   - `sign(NaN)` is `NaN`

/**
 * Returns the smaller of two values.
 * \* \* Special cases:\n * - `NaN.sign` is `NaN`
 * @see abs function
 */
@SinceKotlin("1.2")
public expect fun sign(x: Double): Double

/**
 * Returns the greater of two values.
 * \* \* Special cases:\n * - `NaN.sign` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun max(a: Double, b: Double): Double

/**
 * Raises this value to the power \( \lvert x \rvert \).
 * \* \* Special cases:\n * - \( b.pow(0.0) = 1.0 \)
 */
@SinceKotlin("1.2")
public expect fun pow(x: Double): Double

/**
 * Returns the absolute value of this value.
 * \* \* Special cases:\n * - `NaN.absValue` is `NaN`
 * @see abs function
 */
@SinceKotlin("1.2")
public expect val abs: Double

/**
 * Returns the sign of this value:
 * \* \* Special case:
 *   - `NaN.sign` is `NaN`
 */
@SinceKotlin("1.2")
public expect val sign: Double

/**
 * Returns this value with the sign bit same as of the [sign] value.
 * \* \* If [sign] is `NaN` the sign of the result is undefined.
 * \* @see abs function
 */
@SinceKotlin("1.2")
public expect fun withSign(sign: Double): Double

/**
 * Returns the ulp (unit in the last place) of this value.
 * \* \* Special Cases:
 *   - `NaN.ulp` is `NaN`
 *   - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`
 */
@SinceKotlin("1.2")
public expect val ulp: Double

/**
 * Rounds this value to the nearest integer and converts the result to `Int`.
 * \* \* Special cases:\n * - `x.roundToInt()` is `NaN` if either \( x \) or \( \text{y} \) are `NaN`
 */
@SinceKotlin("1.2")
public expect fun roundToInt(): Int

/**
 * Computes the sine of the angle \( x \) given in radians.
 * \* \* Special cases:\n * - `sin(NaN+Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun sin(x: Float): Float

/**
 * Computes the cosine of the angle \( x \) given in radians.
 * \* \* Special cases:\n * - `cos(NaN+Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun cos(x: Float): Float

/**
 * Computes the tangent of the angle \( x \) given in radians.
 * \* \* Special cases:\n * - `tan(NaN+Inf)` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun tan(x: Float): Float

/**
 * Computes the arc sine of the value \( x \) in the range from `-\( PI/2 \)` to `\( PI/2 \)` radians.
 * \* \* Special cases:\n * - `asin(x)` is `NaN` when \( abs(x) > 1 \)
 */
@SinceKotlin("1.2")
public expect fun asin(x: Float): Float

/**
 * Computes the arc cosine of the value \( x \) in the range from `0.0` to `\( 1 \)` radians.
 * \* \* Special cases:\n * - `acos(x)` is `NaN` when \( abs(x) > 1 \)
 */
@SinceKotlin("1.2")
public expect fun acos(x: Float): Float
Open Source Used in DNAC Platform Arcia

```
produce more precise result for $x$ near zero."

* Special cases:
  - `ln1p(NaN)` is `NaN`
  - `ln1p(x)` is

/**
 * Computes `ln(a + 1)`.
 *
 * This function can be implemented to produce more precise result for $x$ near zero.
 *
 * Special cases:
 *   - `ln(NaN)` is `NaN`
 *   - `ln(x)` is `NaN` when $x < 0.0$ or $x$ is `NaN`

/**
 * Computes the common logarithm (base 10) of the value $x$.
 *
 * @see [ln] function for special cases.
 */
@SinceKotlin("1.2") public expect fun log10(x: Float):

/**
 * Computes the natural logarithm (base `E`) of the value $x$.
 *
 * This function can be implemented to produce more precise result for $x$ near zero.
 *
 * Special cases:
 *   - `ln1p(NaN)` is `NaN`
 *   - `ln1p(x)` is `NaN` where $x < -1.0`
 *   - `ln1p(-1.0)` is `-Inf`
 *   - `ln1p(+Inf)` is `+Inf`
 */
@SinceKotlin("1.2") public expect fun log1p(x: Float):
```
Open Source Used In DNAC Platform Arica 13516

```kotlin
/**
 * Rounds the given value \([x]\) to an integer towards positive infinity.\n * @return the smallest Float value that is greater than or equal to the given value \([x]\) and is a mathematical integer.\n * Special cases:\n * - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `ceil(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `ceil(x)` returns the smaller of two values.\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * @throws IllegalArgumentException when this value is `NaN`\n * @throws IllegalArgumentException when this value is `NaN`\n * @throws IllegalArgumentException when this value is `NaN`\n * @since Kotlin("1.2")\n * public expect fun ceil(x: Float): Float

/**
 * Rounds the given value \([x]\) to an integer towards zero.\n * @return the value \([x]\) having its fractional part truncated.\n * Special cases:\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `truncate(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * @throws IllegalArgumentException when this value is `NaN`\n * @since Kotlin("1.2")\n * public expect fun truncate(x: Float): Float

/**
 * Rounds the given value \([x]\) towards the closest integer with ties rounded towards even integer.\n * Special cases:\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * - `round(x)` is `x` where `x` is `NaN` or `+Inf` or `-Inf` or already a mathematical integer.\n * @since Kotlin("1.2")\n * public expect fun round(x: Float): Float

/**
 * Returns the absolute value of the given value \([x]\).\n * Special cases:\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * See the other overload of [pow] for details.\n * @since Kotlin("1.2")\n * public expect fun abs(x: Float): Float

/**
 * Returns the smallest of two values.\n * @param a the first value\n * @param b the second value\n * @since Kotlin("1.2")\n * public expect fun min(a: Float, b: Float): Float

/**
 * Returns the greater of two values.\n * @param a the first value\n * @param b the second value\n * @since Kotlin("1.2")\n * public expect fun max(a: Float, b: Float): Float

/**
 * Returns the absolute value of the given value \([x]\).\n * Special cases:\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * - `abs(NaN)` is `NaN`\n * @see absoluteValue extension property for [Float]\n * @since Kotlin("1.2")\n * public expect fun abs(n: Int): Int

/**
 * Returns the smaller of two values.\n * @param a the first value\n * @param b the second value\n * @since Kotlin("1.2")\n * public expect fun min(a: Int, b: Int): Int

/**
 * Returns the greater of two values.\n * @param a the first value\n * @param b the second value\n * @since Kotlin("1.2")\n * public expect fun max(a: Int, b: Int): Int

// extensions

/**
 * Returns the sign of this value:\n * - `-1.0` if the value is negative,\n * - zero if the value is zero,\n * - `1.0` if the value is positive.\n * Special cases:\n * - `sign(NaN)` is `NaN`\n * - `sign(NaN)` is `NaN`\n * - `sign(NaN)` is `NaN`\n * @since Kotlin("1.2")\n * public expect fun sign(x: Float): Float

/**
 * Returns this value with the sign bit same as of the [sign] value.\n * @param sign the sign value\n * @since Kotlin("1.2")\n * public expect fun withSign(sign: Float): Float

/**
 * Returns this value with the sign bit same as of the [sign] value.\n * @param sign the sign value\n * @since Kotlin("1.2")\n * public expect fun withSign(sign: Int): Float

/**
 * Returns this value with the sign bit same as of the [sign] value.\n * @param sign the sign value\n * @since Kotlin("1.2")\n * public expect fun withSign(sign: Int): Float

// endregion

// region

```

```
@SinceKotlin("1.2")
public expect fun max(a: Int, b: Int): Int

/**
 * Returns the absolute value of this
 */
@SinceKotlin("1.2")
public expect val Int.absoluteValue: Int

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 */
@SinceKotlin("1.2")
public expect val Int.sign: Int

/**
 * Returns the absolute value of the given value [n].
 */
@SinceKotlin("1.2")
public expect val Long.absoluteValue: Long

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 */
@SinceKotlin("1.2")
public expect val Long.sign: Int


Open Source Used In DNAC Platform Articles
Open Source Used In DNAC Platform Arica  13553
Open Source Used In DNAC Platform Arica  13567
Open Source Used In DNAC Platform

Arica
Open Source Used In DNAC Platform

- Arica
- UAAe,
- OAAO,
- C;QACtB,
- IAAl,
- MAAM,
- C;QACi,
- QAAyB,
- QAAO,
- I;MAChC,c;QAAS,
- O;MACvB,
- IAAl,
- CAAC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- O;MACnB,
- IAAl,
- CAAC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MACChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
- I;MAChC,
- QAAS,
- UAAd,
- C;QAAyB,
- QAAO,
Open Source Used In DNAC Platform Articles

13608
Open Source Used In DNAC Platform Arica
Open Source Used In DNAC Platform Arcs
Open Source Used In DNAC Platform: Arica
1.2741 picocolors 1.0.0

1.2741.1 Available under license:

ISC License

Copyright (c) 2021 Alexey Raspopov, Kostiantyn Denysov, Anton Verinov

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2742 netty-codec-http 4.1.42.Final

1.2742.1 Available under license:

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
#   http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-jar/io/netty/handler/codec/http/HttpScheme.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/HttpClientUpgradeHandler.java
*/
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/HttpServerUpgradeHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// (BSD License: http://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// * copyright notice, this list of conditions and the
// * following disclaimer.
// * Redistributions in binary form must reproduce the above
// * following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocket/WebSocket08FrameEncoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocket/WebSocket13FrameEncoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocket/WebSocket07FrameEncoder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *  http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// (BSD License: http://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-jar/io/netty/handler/codec/http/websocketx/WebSocket08FrameDecoder.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/HttpServerKeepAliveHandler.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/multipart/FileUploadUtil.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketChunkedInput.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

/*
* Adaptation of http://bjoern.hoehrmann.de/utf-8/decoder/dfa/
*
* Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software
* and associated documentation files (the "Software"), to deal in the Software without restriction,
* including without limitation the rights to use, copy, modify, merge, publish, distribute,
* sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
* BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
* FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
* ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/Utf8Validator.java
No license file was found, but licenses were detected in source scan.

/*
Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshaker08.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocket00FrameDecoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/extensions/WebSocketExtensionFilter.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketDecoderConfig.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketServerHandshaker.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
* 2.0 (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/cors/CorsConfigBuilder.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/ReadOnlyHttpHeaders.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/websocketx/WebSocketScheme.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
jar/io/netty/handler/codec/http/HttpServerExpectContinueHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/HttpClientCodec.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/ObjectEncoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractDiskHttpData.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketUtil.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/HttpClientCodec.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/ObjectEncoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/multipart/AbstractDiskHttpData.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/WebSocketUtil.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/HttpClientCodec.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/ObjectEncoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/multipart/InterfaceHttpData.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/PongWebSocketFrame.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/PongWebSocketFrame.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/ContinuationWebSocketFrame.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-
  jar/io/netty/handler/codec/http/websocketx/ContinuationWebSocketFrame.java
jar/io/netty/handler/codec/http/HttpObjectAggregator.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-jar/io/netty/handler/codec/spdy/SpdyHeaderBlockRawDecoder.java
* /opt/cola/permits/1258272149_1643050035.02/0/netty-codec-http-4-1-42-final-sources-jar/io/netty/handler/codec/spdy/SpdyFrameDecoderDelegate.java
1.2743 asm 5.2
1.2744 jaxb-core 2.3.1

1.2744.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
Version 1.0

* 

1. Definitions.

o

1.1. ?Contributor? means each individual or entity that creates or contributes to the creation of Modifications.

o

1.2. ?Contributor Version? means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

o

1.3. ?Covered Software? means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

o

1.4. ?Executable? means the Covered Software in any form other than Source Code.

o

1.5. ?Initial Developer? means the individual or entity that first makes Original Software available under this License.

o

1.6. ?Larger Work? means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

o

1.7. ?License? means this document.

o

1.8. ?Licensable? means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

o

1.9. ?Modifications? means the Source Code and Executable form of any of the following:

+ 

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. ?Original Software? means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. ?Patent Claims? means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. ?Source Code? means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. ?You? (or ?Your?) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, ?You? includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, ?control? means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from
the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the
combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property
claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use,
reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or
portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a
Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that
Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to
make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor
(or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor
Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes
or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor
has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of
Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except
as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the
absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made
available in Source Code form and that Source Code form must be distributed only under the terms of this License.
You must include a copy of this License with every copy of the Source Code form of the Covered Software You
distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable
form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a
medium customarily used for software exchange.
3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients? rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient?s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.
4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as ?Participant?) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY?S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a ?commercial item,? as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of ?commercial computer software? (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and ?commercial computer software documentation? as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it
enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

### 1.2745 twill-core 0.6.0-incubating

1.2745.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-jar/org/apache/twill/internal/kafka/client/IntegerEncoder.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/TwillSpecificationCodec.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/ListenerExecutor.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/FetchedMessage.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/package-info.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/state/MessageCallback.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/EnvKeys.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/SingleRunnableApplication.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/EnvContainerInfo.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/ILoggingEventSerializer.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/ServiceListenerAdapter.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/StackTraceElementCodec.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/package-info.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/BasicFetchedMessage.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/TwillRunResourcesCodec.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/launcher/FindFreePort.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/ProcessLauncher.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/TwillContainerController.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/BrokerService.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/KafkaClient.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/utils/Networks.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/kafka/client/Compression.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/ResourceReportAdapter.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/RuntimeSpecificationCodec.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/JvmOptions.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/utils/Instances.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/kafka/client/SimpleKafkaConsumer.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/ZKMessages.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/ApplicationBundler.java
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/json/LogThrowableCodec.java

No license file was found, but licenses were detected in source scan.

/ *
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1137746303_1614121688.86/0/twill-core-0-6-0-incubating-sources-2-
jar/org/apache/twill/internal/CompositeService.java

1.2746 velocity 2.0
1.2746.1 Available under license :

Apache Velocity - Engine
Copyright 2000-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
your Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2747 pathval 1.1.1

1.2747.1 Available under license :
MIT License

Copyright (c) 2011-2013 Jake Luer jake@alogicalparadox.com

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO
THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE
USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2748 @babel/plugin-transform-dotall-regex

7.16.0

1.2748.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2749 jackson-dataformat-csv 2.10.5

1.2749.1 Available under license:

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1595302282882
Build-Jdk: 1.8.0_162
Built-By: tatu
Bundle-Description: Support for reading and writing CSV-encoded data via Jackson abstractions.
Bundle-DocURL: https://github.com/FasterXML/jackson-dataformats-text
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: Jackson-dataformat-Csv
Bundle-SymbolicName: com.fasterxml.jackson.dataformat.csv
Bundle-Version: 2.10.5
Created-By: Apache Maven Bundle Plugin
Export-Package: com.fasterxml.jackson.dataformat.csv;version="2.10.5";
Implementation-Build-Date: 2020-07-21 03:31:04+0000
Implementation-Title: Jackson-dataformat-Csv
Implementation-Vendor: FasterXML

Open Source Used in DNAC Platform Artica 13733
Implementation-Vendor-Id: com.fasterxml.jackson.dataformat
Implementation-Version: 2.10.5
Import-Package: com.fasterxml.jackson.core;version="[2.10,3)";com.fasterxml.jackson.core.base;version="[2.10,3)";com.fasterxml.jackson.core.format;version="[2.10,3)";com.fasterxml.jackson.core.io;version="[2.10,3)";com.fasterxml.jackson.core.json;version="[2.10,3)";com.fasterxml.jackson.core.type;version="[2.10,3)";com.fasterxml.jackson.core.util;version="[2.10,3)";com.fasterxml.jackson.databind;version="[2.10,3)";com.fasterxml.jackson.databind.cfg;version="[2.10,3)";com.fasterxml.jackson.databind.introspect;version="[2.10,3)";com.fasterxml.jackson.databind.util;version="[2.10,3)";com.fasterxml.jackson.dataformat.csv;version="[2.10,3)";com.fasterxml.jackson.dataformat.csv.impl;version="[2.10,3)"
Private-Package: com.fasterxml.jackson.dataformat.csv.impl
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.7))"
Specification-Title: Jackson-dataformat-CSV
Specification-Vendor: FasterXML
Specification-Version: 2.10.5
Tool: Bnd-4.2.0.201903051501
X-Compile-Source-JDK: 1.7
X-Compile-Target-JDK: 1.7

Found in path(s):
* /opt/cola/permits/1154615200_1618868383.1/0/jackson-dataformat-csv-2-10-5-5-jar/META-INF/MANIFEST.MF

---

**1.2750 junit-platform-commons 1.7.0**

**1.2750.1 Available under license:**

Eclipse Public License - v 2.0

=====================================

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

### 1. Definitions

Contribution means:
* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
* **b)** in the case of each subsequent Contributor:
  * **i)** changes to the Program, and
  * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.
Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

### 2. Grant of Rights

**a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

**b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

**c)** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute
the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

**d)** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

**e)** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

### 3. Requirements

**3.1** If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
  * **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
  * **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
  * **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
  * **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

**3.2** When the Program is Distributed as Source Code:

* **a)** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
* **b)** a copy of this Agreement must be included with each copy of the Program.

**3.3** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

### 4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor
(Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **(a)** promptly notify the Commercial Contributor in writing of such claim, and **(b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

### 5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

### 6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### 7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.
All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

#### Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.
Apache Parquet Hadoop
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.salaranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components (as well their dependencies) may be licensed under different licenses.

To find the details that apply to this artifact see the accompanying LICENSE file.

For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

This copy of Jackson JSON processor `jackson-module-afterburner` module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Additional licensing information exists for following 3rd party library dependencies

### ASM

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2753 ansicolors 0.2.1 0.2.1

1.2753.1 Available under license:

Copyright 2013 Thorsten Lorenz.
All rights reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2754 babel-plugin-transform-es2015-block-scoping 6.26.0

1.2754.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2755 opentracing-util 0.31.0

1.2755.1 Available under license:

No license file was found, but licenses were detected in source scan.

2018 The OpenTracing Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
/*
 * Copyright 2016-2018 The OpenTracing Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

1.2756 css-select 1.2.0 1.2.0

1.2756.1 Available under license:

Copyright (c) Felix Bhm
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.
THIS IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2757 @babel/code-frame 7.16.0

1.2757.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2758 to-buffer 1.1.1 1.1.1

1.2758.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016 Mathias Buus

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2759 safe-json-stringify 1.2.0 1.2.0

1.2759.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2760 jaeger-core 1.5.0

1.2760.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2017, Uber Technologies, Inc
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/HttpBaggageRestrictionManagerProxy.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/http/BaggageRestrictionResponse.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/propagation/HexCodec.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/BaggageSetter.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/Reference.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/utils/Java6CompatibleThreadLocalRandom.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/Restriction.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/DefaultBaggageRestrictionManager.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/BaggageRestrictionManager.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/BaggageRestrictionManagerException.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/baggage/RemoteBaggageRestrictionManager.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/BaggageRestrictionManagerProxy.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019, The Jaeger Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-
jar/io/jaegertracing/internal/propagation/BinaryCodec.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2018, The Jaeger Authors
 * Copyright (c) 2017, Uber Technologies, Inc
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-
jar/io/jaegertracing/internal/propagation/B3TextMapCodec.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016-2017, Uber Technologies, Inc
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (c) 2018, The Jaeger Authors
* Copyright (c) 2016, Uber Technologies, Inc
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/JaegerSpan.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/JaegerTracer.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/propagation/TextMapCodec.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/package-

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (c) 2018, The Jaeger Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/package-
info.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/senders/NoopSenderFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/exceptions/TraceIdOutOfBounds.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/JaegerObjectFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/senders/NoopSender.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/spi/SenderFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/PropagationRegistry.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/senders/SenderResolver.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2016, Uber Technologies, Inc
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/samplers/RateLimitingSampler.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/samplers/ConstSampler.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/utils/Http.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/exceptions/UnsupportedFormatException.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/exceptions/EmptyIpException.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/internal/exceptions/SamplingStrategyErrorException.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1.5-0-sources-jar/io/jaegertracing/spi/Extractor.java
jar/io/jaegertracing/internal/Constants.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/samplers/http/RateLimitingSamplingStrategy.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/Sender.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/metrics/Metrics.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/clock/SystemClock.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/samplers/ProbabilisticSampler.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/metrics/InMemoryMetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/metrics/NoopMetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/MetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/propagation/CompositeCodec.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2017, The Jaeger Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/Codec.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/metrics/InMemoryMetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/metrics/NoopMetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/spi/MetricsFactory.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/propagation/CompositeCodec.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2020, The Jaeger Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/clock/MillisAccurateClock.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/MDCScopeManager.java
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/clock/MicrosAccurateClock.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020, OpenTelemetry Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1152086972_1618069762.04/0/jaeger-core-1-5-0-sources-jar/io/jaegertracing/internal/propagation/TraceContextCodec.java
1.2761.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2762 to-fast-properties 1.0.3 1.0.3
1.2762.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Petka Antonov
2015 Sindre Sorhus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2763 isarray 0.0.1 0.0.1

1.2763.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2764 to-fast-properties 2.0.0

1.2764.1 Available under license:

MIT License

Copyright (c) 2014 Petka Antonov
              2015 Sindre Sorhus

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2765 object-inspect 1.10.3 1.10.3

1.2765.1 Available under license :

MIT License

Copyright (c) 2013 James Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2766 async 1.4.1

1.2766.1 Available under license :

The MIT License (MIT)

Copyright (c) Spatie bvba <info@spatie.be>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2767 jetty-http 9.3.27.v20190418

1.2767.1 Available under license:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which:

(i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and

(ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.
2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY
CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
1.2768 parseqs 0.0.6 0.0.6
1.2768.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Gal Koren

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2769 package-json 6.5.0
1.2769.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
1.2770 babel-helper-function-name 6.24.1

6.24.1

1.2770.1 Available under license :

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2771 acorn 2.7.0

1.2771.1 Available under license :

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2772 resolve-url 0.2.1 0.2.1

1.2772.1 Available under license:
The MIT License (MIT)

Copyright (c) 2013 Simon Lydell

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2773 strip-ansi 5.2.0

1.2773.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2774 requires-port 1.0.0 1.0.0
1.2774.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015 Unshift.io, Arnout Kazemier, the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2775 password-hash 1.2.2 1.2.2
1.2775.1 Available under license :
Copyright (c) 2011 David Wood <bitprobe@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2776 websocket-api 9.3.27.v20190418

1.2777 glob 7.1.6 7.1.6
1.2777.1 Available under license :
   The ISC License

   Copyright (c) Isaac Z. Schlueter and Contributors

   Permission to use, copy, modify, and/or distribute this software for any
   purpose with or without fee is hereby granted, provided that the above
   copyright notice and this permission notice appear in all copies.

   THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
   WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
   MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
   ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
   WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
   ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
   IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

   ## Glob Logo

   Glob's logo created by Tanya Brassie <http://tanyabrassie.com/>, licensed
   under a Creative Commons Attribution-ShareAlike 4.0 International License
   https://creativecommons.org/licenses/by-sa/4.0/

1.2778 commons-email 1.5
1.2778.1 Available under license :
   Apache Commons Email
   Copyright 2001-2017 The Apache Software Foundation

   This product includes software developed at
   The Apache Software Foundation (http://www.apache.org/).

   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

W ITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2779 qs 6.5.1
1.2779.1 Available under license:
Copyright (c) 2014 Nathan LaFreniere and other contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* The names of any contributors may not be used to endorse or promote
  products derived from this software without specific prior written
  permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS AND CONTRIBUTORS BE LIABLE
FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* * *

The complete list of contributors can be found at: https://github.com/hapijs/qs/graphs/contributors

1.2780 babel-types 6.26.0
1.2780.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The Software identified by reference to the Ecma Standard* ("Software") is protected by copyright and is
being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE
ECMA
CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTicular PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
1.2781 mime 2.4.4 2.4.4
1.2781.1 Available under license :

The MIT License (MIT)

Copyright (c) 2010 Benjamin Thomas, Robert Kieffer

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2782 org.jacoco.agent.rt 0.7.9
1.2782.1 Available under license :

License

=======

Copyright (c) 2009, 2017 Mountainminds GmbH & Co. KG and Contributors

The JaCoCo Java Code Coverage Library and all included documentation is made
available by Mountainminds GmbH & Co. KG, Munich. Except indicated below, the
Content is provided to you under the terms and conditions of the Eclipse Public
License Version 1.0 ("EPL"). A copy of the EPL is available at

Please visit
for the complete license information including third party licenses and trademarks.
1.2783 encodeurl 1.0.2 1.0.2

1.2783.1 Available under license:
(The MIT License)

Copyright (c) 2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2784 fn-name 3.0.0 3.0.0

1.2784.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
1.2785 avahi 0.6.32~rc+dfsg-1ubuntu2.3

1.2785.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations
below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for
making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.
If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full
notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or
your school, if any, to sign a "copyright disclaimer" for the library,
if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James
Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.2786 jheaps 0.9
1.2786.1 Available under license :
No license file was found, but licenses were detected in source scan.

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
<modelVersion>4.0.0</modelVersion>

<groupId>org.jheaps</groupId>
<artifactId>jheaps</artifactId>
<version>0.9</version>
<organizationUrl>https://github.com/d-michail</organizationUrl>
</developer>
</developers>

<repositories>
<repository>
<id>sonatype-nexus-snapshots</id>
<name>Sonatype Nexus Snapshots</name>
<url>https://oss.sonatype.org/content/repositories/snapshots</url>
<releases>
<enabled>false</enabled>
</releases>
<snapshots>
<enabled>true</enabled>
</snapshots>
</repository>
</repositories>

<distributionManagement>
<snapshotRepository>
<id>ossrh</id>
<url>https://oss.sonatype.org/content/repositories/snapshots/</url>
</snapshotRepository>
<repository>
<id>ossrh</id>
<url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
</repository>
</distributionManagement>

<dependencies>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>${junit.version}</version>
<scope>test</scope>
</dependency>
</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-compiler-plugin</artifactId>
<version>${maven-compiler-plugin.version}</version>
<configuration>
<source>${java.version}</source>
<target>${java.version}</target>
</configuration>
</plugin>
</plugins>
</build>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-jar-plugin</artifactId>
.VERSION>${maven-jar-plugin.version}</VERSION>
<configuration>
<excludes>
<exclude>logback.xml</exclude>
</excludes>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-release-plugin</artifactId>
.VERSION>${maven-release-plugin.version}</VERSION>
<configuration>
<mavenExecutorId>forked-path</mavenExecutorId>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
.VERSION>${maven-javadoc-plugin.version}</VERSION>
<configuration>
<show>public</show>
>windowtitle>JHeaps</windowtitle>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-deploy-plugin</artifactId>
.VERSION>${maven-deploy-plugin.version}</VERSION>
<executions>
<execution>
<id>default-deploy</id>
<phase>deploy</phase>
<goals>
<goal>deploy</goal>
</goals>
</execution>
</executions>
</plugin>
<groupId>org.jacoco</groupId>
<artifactId>jacoco-maven-plugin</artifactId>
<version>${jacoco.version}</version>
<executions>
  <execution>
    <id>default-prepare-agent</id>
    <goals>
      <goal>prepare-agent</goal>
    </goals>
  </execution>
  <execution>
    <id>default-report</id>
    <phase>prepare-package</phase>
    <goals>
      <goal>report</goal>
    </goals>
  </execution>
  <execution>
    <id>default-check</id>
    <goals>
      <goal>check</goal>
    </goals>
    <configuration>
      <rules>
        <rule>
          <element>BUNDLE</element>
          <limits>
            <limit>
              <counter>INSTRUCTION</counter>
              <value>COVEREDRATIO</value>
              <minimum>0.75</minimum>
            </limit>
          </limits>
        </rule>
        <rule>
          <counter>CLASS</counter>
          <value>COVEREDRATIO</value>
          <minimum>0.75</minimum>
        </rule>
        <rule>
          <counter>METHOD</counter>
          <value>COVEREDRATIO</value>
          <minimum>0.75</minimum>
        </rule>
        <rule>
          <counter>BRANCH</counter>
          <value>COVEREDRATIO</value>
          <minimum>0.75</minimum>
        </rule>
      </rules>
    </configuration>
  </execution>
</executions>
<counter>COMPLEXITY</counter>
<value>COVEREDRATIO</value>
<minimum>0.75</minimum>
</limit>
</counter>
<counter>LINE</counter>
<value>COVEREDRATIO</value>
<minimum>0.75</minimum>
</limit>
</limits>
</rule>
</configuration>
</execution>
</executions>
</plugin>

</plugins>
</build>

<reporting>
<plugins>
<plugin>
<groupId>org.jacoco</groupId>
<artifactId>jacoco-maven-plugin</artifactId>
<reportSets>
<reportSet>
<reports>
<report>report</report>
</reports>
</reportSet>
</reportSets>
</plugin>
</plugins>
</reporting>

<profiles>
<profile>
<id>release-sign-artifacts</id>
<activation>
<property>
<name>performRelease</name>
<value>true</value>
</property>
</activation>
</profile>
</profiles>

<build>
<plugins>
</plugins>
</build>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-gpg-plugin</artifactId>
<version>1.5</version>
<executions>
  <execution>
    <id>sign-artifacts</id>
    <phase>verify</phase>
    <goals>
      <goal>sign</goal>
    </goals>
  </execution>
</executions>
</plugin>

<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-javadoc-plugin</artifactId>
  <configuration>
    <show>public</show>
    <windowtitle>JHeaps</windowtitle>
  </configuration>
  <executions>
    <execution>
      <id>attach-javadocs</id>
      <goals>
        <goal>jar</goal>
      </goals>
    </execution>
  </executions>
</plugin>

<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-source-plugin</artifactId>
  <executions>
    <execution>
      <id>attach-sources</id>
      <goals>
        <goal>jar-no-fork</goal>
      </goals>
    </execution>
  </executions>
</plugin>

<build>
  <plugins>
    <profile>
      <id>checkstyle</id>
    </profile>
  </plugins>
</build>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-checkstyle-plugin</artifactId>
  <version>${maven-checkstyle-plugin.version}</version>
  <dependencies>
    <dependency>
      <groupId>com.puppycrawl.tools</groupId>
      <artifactId>checkstyle</artifactId>
      <version>${checkstyle.version}</version>
    </dependency>
  </dependencies>
  <executions>
    <execution>
      <id>checkstyle</id>
      <phase>validate</phase>
      <goals>
        <goal>check</goal>
      </goals>
    </execution>
  </executions>
  <configuration>
    <configLocation>etc/jheaps_checks.xml</configLocation>
    <linkXRef>false</linkXRef>
  </configuration>
</plugin>

Found in path(s):
* /opt/cola/permits/1179722343_1626156109.63/0/jheaps-0-9-jar/META-INF/maven/org.jheaps/jheaps/pom.xml

1.2787 google-http-client 1.20.0

1.2788 jersey-servlet 1.19
1.2789 pkg-dir 4.2.0

1.2789.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2790 sprintf-js 1.1.0 1.1.0

1.2790.1 Available under license:
Copyright (c) 2007-2016, Alexandru Marasteanu <hello [at] alexei (dot) ro>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of this software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2791 babel-plugin-transform-es2015-template-literals 6.22.0 6.22.0
1.2791.1 Available under license:
   The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2792 json-buffer 3.0.0
1.2792.1 Available under license:
   Copyright (c) 2013 Dominic Tarr

Permission is hereby granted, free of charge,
to any person obtaining a copy of this software and
associated documentation files (the "Software"), to
deal in the Software without restriction, including
without limitation the rights to use, copy, modify,
merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom
the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice
shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2793 slf4j 1.7.26
1.2793.1 Available under license :
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
<title>SLF4J License</title>
<link rel="stylesheet" type="text/css" media="screen" href="css/site.css" />
</head>
<body>
<script type="text/javascript">prefix='';</script>
<script src="templates/header.js" type="text/javascript"></script>
<div id="left">
<script src="templates/left.js" type="text/javascript"></script>
</div>
<div id="right">
<script src="templates/right.js" type="text/javascript"></script>
</div>
<div id="content">

<h1>Licensing terms for SLF4J</h1>
<p>SLF4J source code and binaries are distributed under the
MIT license.
</p>
<div class="source">
Copyright (c) 2004-2017 QOS.ch
All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including

Open Source Used In DNAC Platform Arica 13801


without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

These terms are identical to those of the MIT License, also called the X License or the X11 License, which is a simple, permissive non-copyleft free software license. It is deemed compatible with virtually all types of licenses, commercial or otherwise. In particular, the Free Software Foundation has declared it compatible with GNU GPL. It is also known to be approved by the Apache Software Foundation as compatible with the Apache Software License.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2794 shebang-regex 1.0.0 1.0.0

1.2794.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.2795 snapdragon-node 2.1.1 2.1.1
1.2795.1 Available under license:
The MIT License (MIT)

Copyright (c) 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2796 log4j-slf4j-impl 2.14.0
1.2796.1 Available under license:
Apache Log4j SLF4J Binding
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2797 debug 3.2.6 3.2.6

1.2797.1 Available under license :
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2798 electron-to-chromium 1.3.690

1.2798.1 Available under license :
Copyright 2018 Kilian Valkhof

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2799 papa-parse 5.1.1 5.1.1
1.2799.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015 Matthew Holt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2800 regenerate-unicode-properties 9.0.0
1.2800.1 Available under license:

Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2801 merge-descriptors 1.0.1 1.0.1
1.2801.1 Available under license:
(The MIT License)

Copyright (c) 2013 Jonathan Ong <me@jongleberry.com>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2802 finalhandler 0.4.1 0.4.1
1.2802.1 Available under license:
(The MIT License)

Copyright (c) 2014-2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND.
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2803 handlebars 4.7.6 4.7.6

1.2803.1 Available under license:
Copyright (C) 2011-2019 by Yehuda Katz

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2804 get-symbol-description 1.0.0

1.2804.1 Available under license:
MIT

1.2805 common-prefix 1.1.0 1.1.0

1.2805.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2806 jsreport-import-export 2.1.1 2.1.1

1.2806.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2807 babel-plugin-transform-object-rest-spread 6.26.0

1.2807.1 Available under license:

Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being
made available under the "BSD License", included below. This Software may be subject to third party rights (rights
from parties other than Ecma International), including patent rights, and no licenses under such third party rights
are granted under this license even if the third party concerned is a member of Ecma International. SEE THE

ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-
international.org/memento/codeofconduct.htm FOR
INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO
IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived
   from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports

Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copyrights noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
1.2808 lodash-filter 4.6.0 4.6.0

1.2808.1 Available under license:

Copyright jQuery Foundation and other contributors <https://jquery.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2809 parquet-format 2.4.0

1.2809.1 Available under license:

Apache Parquet Format
Copyright 2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product depends on SLF4J and includes SLF4J in this binary artifact. SLF4J is a simple logging facade for Java.

Copyright: 2004-2013 QOS.ch.
Home page: http://www.slf4j.org/
License: http://slf4j.org/license.html (MIT license)

The following is the SLF4J license (MIT):

Copyright (c) 2004-2013 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product depends on Apache Thrift and includes it in this binary artifact.

Copyright: 2006-2010 The Apache Software Foundation.
Home page: https://thrift.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

1.2810 es6-promise 4.2.8 4.2.8
1.2810.1 Available under license :
Copyright (c) 2014 Yehuda Katz, Tom Dale, Stefan Penner and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2811 babel-helper-explode-class 6.24.1
1.2811.1 Available under license:

Apache License
Version 2.0, January 2011
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

MIT License

Copyright (c) 2014-2017 Sebastian McKenzie <sebmck@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2015 Sebastian McKenzie <sebmck@gmail.com>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2812 is-finite 1.0.2 1.0.2
1.2812.1 Available under license:

The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2813 parquet-common 1.10.0

1.2813.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
  Licensed to the Apache Software Foundation (ASF) under one
  or more contributor license agreements. See the NOTICE file
  distributed with this work for additional information
  regarding copyright ownership. The ASF licenses this file
  to you under the Apache License, Version 2.0 (the
  "License"); you may not use this file except in compliance
  with the License. You may obtain a copy of the License at
  *
  * http://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing,
  * software distributed under the License is distributed on an
  * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
  * KIND, either express or implied. See the License for the
  * specific language governing permissions and limitations
  * under the License.
  */

Found in path(s):
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/bytes/SingleBufferInputStream.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/ShouldNeverHappenException.java
  */opt/ cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/SemanticVersion.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/Strings.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/Files.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/IOExceptionUtils.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/bytes/ConcatenatingByteArrayCollector.java
  */opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/util/DynConstructors.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/glob/WildcardPath.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/glob/GlobNode.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/VersionParser.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/bytes/ByteBufferAllocator.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/glob/GlobExpander.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/util/DynMethods.java

No license file was found, but licenses were detected in source scan.

/ *
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/bytes/HeapByteBufferAllocator.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/Preconditions.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/bytes/CapacityByteArrayOutputStream.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
jar/org/apache/parquet/Ints.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
No license file was found, but licenses were detected in source scan.

/**
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-jar/org/apache/parquet/OutputStreamCloseException.java
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 */
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/Exceptions.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/PositionOutputStream.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/hadoop/metadata/CompressionCodecName.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/DelegatingSeekableInputStream.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/InputFile.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/OutputFile.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/SeekableInputStream.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/io/DelegatingPositionOutputStream.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/hadoop/codec/CompressionCodecNotSupportedException.java
* /opt/cola/permits/1136206342_1613708930.97/0/parquet-common-1-10-0-sources-1-
  jar/org/apache/parquet/compression/CompressionCodecFactory.java

1.2814 request-promise 4.2.6 4.2.6
1.2814.1 Available under license:

ISC License

Copyright (c) 2020, Nicolai Kamenzky, Ty Abonil, and contributors

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2815 es6-promise 3.2.1 3.2.1
1.2815.1 Available under license:
Copyright (c) 2014 Yehuda Katz, Tom Dale, Stefan Penner and contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2816 esrecurse 4.3.0
1.2816.1 Available under license:
No license file was found, but licenses were detected in source scan.

// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are met:
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
Open Source Used In DNAC Platform Arica  13842

Found in path(s):
* /opt/cola/permits/1162566273_1620862144.53/0/estools-esrecurse-v4-3-0-0-g8ceb637-1-tar-gz/estools-esrecurse-f285b57/test/traverse.js
* /opt/cola/permits/1162566273_1620862144.53/0/estools-esrecurse-v4-3-0-0-g8ceb637-1-tar-gz/estools-esrecurse-f285b57/gulpfile.babel.js
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Yusuke Suzuki <utatane.tea@gmail.com>
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 *  * Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *  * Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
 * DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):
* /opt/cola/permits/1162566273_1620862144.53/0/estools-esrecurse-v4-3-0-0-g8ceb637-1-tar-gz/estools-esrecurse-f285b57/esrecurse.js
No license file was found, but licenses were detected in source scan.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

Found in path(s):
* /opt/cola/permits/1162566273_1620862144.53/0/estools-esrecurse-v4-3-0-0-g8ceb637-1-tar-gz/estools-esrecurse-f285b57/README.md
1.2817 toposort 2.0.2 2.0.2

1.2817.1 Available under license:

Toposort - Topological sorting for node.js
Copyright (c) 2012 by Marcel Klehr <mklehr@gmx.net>
MIT LICENSE
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2818 velocity-engine-core 2.0

1.2818.1 Available under license:

Apache Velocity - Engine
Copyright 2000-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache Commons IO
Copyright 2002-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"...
means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2819 avro 1.7.6
1.2819.1 Available under license :

Apache Avro
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2820 istanbul-lib-coverage 3.0.0

1.2820.1 Available under license:

Copyright 2012-2015 Yahoo! Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL YAHOO! INC. BE LIABLE FOR ANY
DIREC T, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUD ING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2821 sqlline 1.9.0

1.2821.1 Available under license:
No license file was found, but licenses were detected in source scan.

*/
// Licensed to Julian Hyde under one or more contributor license
// agreements. See the NOTICE file distributed with this work for
// additional information regarding copyright ownership.

// Julian Hyde licenses this file to you under the Modified BSD License
// (the "License"); you may not use this file except in compliance with
// the License. You may obtain a copy of the License at:

// http://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/Application.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/ReflectiveCommandHandler.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/Dialect.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/DispatchCallback.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/BuiltinHighlightStyle.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/ClassNameCompleter.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/BufferedReader.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SqlLineOpts.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SqlLineHighlighter.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/TableNameCompleter.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SeparatedValuesOutputFormat.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/JsonOutputFormat.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/DatabaseConnection.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SqlCompleter.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SqlLine.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/VerticalOutputFormat.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/AttributedStyles.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/SqlLineParser.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/BuiltinDialect.java
*/opt/cola/permits/1179408911_1626107141.97/0/sqlline-1-9-0-sources-jar/sqlline/XmlAttributeOutputFormat.java
No license file was found, but licenses were detected in source scan.

SQLLine 1.8.0

Copyright 2006-2019 Julian Hyde

SQLLine Manual
Introduction
License and Terms of Use
Obtaining SQLLine
Installation
Required software
Installing SQLLine
Using SQLLine
Running SQLLine
Connecting to a database
Issuing SQL commands
Command Completion
Command Reference
all
all — Execute SQL against all active connections
appconfig
appconfig - Sets application configuration class name
autocommit
autocommit — Enable or disable automatic transaction commit mode
batch
batch — Start or execute a batch of SQL statements
brief
brief — Enable terse output mode
close
close — Close the active connection
columns
columns — Display columns of a table
commandhandler
commandhandler — Adds command handlers with commands
commit
commit — Commit the current transaction
connect
connect — Connect to a database
dbinfo
dbinfo — Provide information about the current database
describe
describe — Describe a table
dropall
dropall — Drop all tables in the database or in the schema
exportedkeys
exportedkeys — List exported foreign keys for a database
go
go — Change to a different active connection
help
help — Display help information
history
history — Display the command history
-d Print timestamps for each event (default for !history)
-f Print full time date stamps in the US format
-E Print full time date stamps in the European format
-i Print full time date stamps in ISO8601 format
-n Suppresses command numbers
-r Reverses the order of the commands
importedkeys
importedkeys — List imported foreign keys for a database
indexes
indexes — Display indexes for a table
isolation
isolation — Set the transaction isolation mode for the active connection
list
list — Display all active connections
metadata
metadata — Invoke arbitrary metadata commands
nickname
nickname - Create a friendly name for the connection (updates command prompt)
outputformat
outputformat — Change the method for displaying SQL results
primarykeys
primarykeys — Display the primary key columns for a table
procedures
procedures — List stored procedures
prompthandler
prompthandler - Set custom prompt handler class name
properties
properties — Connect to the database defined in the specified properties file.
quit
quit — Exit SQLLine
readonly
readonly - Enable or disable readonly connection
reconnect
reconnect — Reconnect to the current database
record
record — Begin recording all output from SQL commands
rehash
rehash — Obtain a list of all tables and columns from the database
rollback
rollback — Roll back the current transaction
run
run — Execute a command script
save
save — Save the current preferences
scan
scan — Scan class path for JDBC drivers
schemas
schemas — List all the schemas in the database
script
script — Save executed commands to a file
set
set — Set a preference
sql
sql — Execute a SQL against a database
tables
tables — List all the tables in the database
verbose
verbose — Enable verbose output
Parameter Reference
autocommit
autosave
color
colorScheme
colorScheme
confirm
confirmPattern
csvDelimiter
csvQuoteCharacter
dateformat
escapeOutput
fastconnect
force
headerinterval
historyfile
incremental
isolation
maxcolumnwidth
maxHistoryFileRows
maxHistoryRows
maxwidth
mode
nullValue
numberformat
outputformat
prompt
rightprompt
rowlimit
showheader
shownestederrs
showtime
showwarnings
strictJdbc
silent
timeformat
timestampformat
trimscripts
uselinecontinuation
verbose
JDBC Driver Support
Project Information
SQLLine Manual

Table of Contents

Introduction
License and Terms of Use
Obtaining SQLLine
Installation
Required software
Installing SQLLine
Using SQLLine
Running SQLLine  
Connecting to a database  
Issuing SQL commands  
Command Completion  
Command Reference  
all  
all — Execute SQL against all active connections  
appconfig  
appconfig - Sets application configuration class name  
autocommit  
autocommit — Enable or disable automatic transaction commit mode  
batch  
batch — Start or execute a batch of SQL statements  
brief  
brief — Enable terse output mode  
close  
close — Close the active connection  
columns  
columns — Display columns of a table  
commandhandler  
commandhandler — Adds command handlers with commands  
commit  
commit — Commit the current transaction  
connect  
connect — Connect to a database  
dbinfo  
dbinfo — Provide information about the current database  
describe  
describe — Describe a table  
dropall  
dropall — Drop all tables in the database or in the schema  
exportedkeys  
exportedkeys — List exported foreign keys for a database  
go  
go — Change to a different active connection  
help  
help — Display help information  
history  
history — Display the command history  
-d Print timestamps for each event (default for !history)  
-f Print full time date stamps in the US format  
-E Print full time date stamps in the European format  
-i Print full time date stamps in ISO8601 format  
-n Suppresses command numbers  
-r Reverses the order of the commands  
importedkeys  
importedkeys — List imported foreign keys for a database  
indexes
indexes — Display indexes for a table
isolation
isolation — Set the transaction isolation mode for the active connection
list
list — Display all active connections
metadata
metadata — Invoke arbitrary metadata commands
outputformat
outputformat — Change the method for displaying SQL results
primarykeys
primarykeys — Display the primary key columns for a table
procedures
procedures — List stored procedures
prompthandler
prompthandler — Set custom prompt handler class name
properties
properties — Connect to the database defined in the specified properties file.
quit
quit — Exit SQLLine
reconnect
reconnect — Reconnect to the current database
record
record — Begin recording all output from SQL commands
rehash
rehash — Obtain a list of all tables and columns from the database
rollback
rollback — Roll back the current transaction
run
run — Execute a command script
save
save — Save the current preferences
schemas
schemas — List all the schemas in the database
scan
scan — Scan class path for JDBC drivers
script
script — Save executed commands to a file
set
set — Set a preference
sql
sql — Execute a SQL against a database
tables
tables — List all the tables in the database
verbose
verbose — Enable verbose output
Parameter Reference
autosave
color
colorScheme
confirm
confirmPattern
csvDelimiter
csvQuoteCharacter
dateformat
escapeOutput
fastconnect
force
headerinterval
historyfile
incremental
isolation
maxcolumnwidth
maxHistoryFileRows
maxHistoryRows
maxwidth
mode
nullValue
numberformat
outputformat
prompt
rightprompt
rowlimit
showheader
shownestederrs
showtime
showwarnings
silent
strictJdbc
timeformat
timestampformat
trimscripts
uselinecontinuation
verbose

JDBC Driver Support

Introduction

SQLLine is a pure-Java console based utility for connecting to relational databases and executing SQL commands. It is similar to other command-line database access utilities like sqlplus for Oracle, mysql for MySQL, and isql for Sybase/SQL Server. Since it is pure-Java, it is platform independent, and will run on any platform that can run Java 1.3 or higher.

License and Terms of Use

SQLLine is distributed under the BSD License, meaning that you are free to redistribute, modify, or sell the software with almost no restrictions. For the text of the license, see the license text.
For information on obtaining the software under another license, contact the copyright holder: mwp1@cornell.edu.

Obtaining SQLLine

SQLLine is hosted on SourceForge, and is located at https://sqlline.sf.net. The latest release can be downloaded from https://sqlline.sf.net/download.html.

Installation

Required software

Installing SQLLine

Required software

SQLLine depends on the following software:

Java™ Virtual Machine - SQLLine is a pure java program, and requires Java version 1.3 or higher in order to run. The latest JVM can be downloaded from https://java.sun.com.

JLine - SQLLine uses the JLine console reader for command line editing, tab-completion, and command history. It can be downloaded from https://jline.sf.net. Version 0.8.1 or higher is required.

JDBC™ Driver(s) - Since SQLLine uses the Java Database Connectivity package to connect to your database, you need to obtain the correct JDBC driver libraries for your database. Any JDBC compliant driver can be used for this purpose. Drivers can be located by referring to the section on known JDBC drivers, or else by referring to your database vendor's web site.

Installing SQLLine

There are currently no set standards for installing Java software on different platforms. The simplest method for installing SQLLine in a platform independant way is as follows:

Create a new directory/folder where you like. This will be referred to as sqllinedir.

Download sqlline.jar into sqllinedir.

Download the latest jline.jar from https://jline.sf.net into sqllinedir.

Download your database's JDBC driver files into sqllinedir. Note that some JDBC drivers require some installation, such as uncompressing or unzipping.

Using SQLLine

Running SQLLine

Connecting to a database

Issuing SQL commands

Command Completion

Running SQLLine
SQLLine can be run with the following command:

```
java {-Djava.ext.dirs=sqllinedir} {sqllinedir/sqlline.jar} [options...] [properties files...]
```

For example, if your sqllinedir is /usr/share/java/ (this is typical for Debian Linux), then you might run:

```
java -Djava.ext.dirs=/usr/share/java/ /usr/share/java/sqlline.jar
```

Once you have started SQLLine, you will a prompt that reads "sqlline> ". From here, you can enter either SQLLine commands (beginning with a "!" character), or a SQL statement (if you are currently connected to a database). For example, entering "!help" will display a list of all the available commands.

Connecting to a database

SQLLine is only useful once you are connected to a database. In order to do so, you need to use the "!connect" command, with the JDBC URL of the database you are going to connect to, followed by the username and password (followed optionally by the driver class name). The JDBC URL is specific to the JDBC driver that you are using for your database, but will always start with "jdbc:", and usually has the machine name of the database and the name of the database itself as part of the string. Quotes (both single and double) could be used in case of url, username, password contain spaces. Consult your driver's documentation for details on the URL to use.

For example, to connect to an Oracle database, you might enter:

```
sqlline> !connect jdbc:oracle:thin:@localhost:1521:mydb scott tiger oracle.jdbc.driver.OracleDriver
```

Connected to: Oracle (version Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production)
Driver: Oracle JDBC driver (version 9.0.1.0.0)
Autocommit status: true

```
0: jdbc:oracle:thin:@localhost:1521:mydb>
```

Once you are connected, the prompt will change the reflect URL of the current connection.

SQLLine can also accept command line options to connect to a database. If a file is specified on the command line, it will be treated as a properties file, and try to connect to the database specified therein. For details, see the properties command documentation.

SQLLine allows you to have multiple connections to databases (either the same database, or a different database) open at the same time. Issuing the connect command multiple times will add the new connection to the list of open connections. The "current" connection (against which all SQL statements will be executed) can be switched with the go command. The currently open connections can be listed with the list command.

Issuing SQL commands
Once you are connected to a database, you can begin SQL statements against the current connection. For example:

Example 1. Issuing SQL commands

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM MYTABLE;

+-----+
| ID  |
+-----+
| 1   |
| 2   |
| 3   |
| 4   |
+-----+
4 rows selected (0.012 seconds)
0: jdbc:oracle:thin:@localhost:1521:mydb>

Command Completion

On UNIX systems, SQLLine allows command-line completion using the tab key. This will be familiar for users of such popular shells as bash and tcsh. In general, hitting the TAB key when part of the way through typing a command will fill in the rest of the command, or else will display a selection of appropriate possibilities.

Command Reference

all
all — Execute SQL against all active connections
appconfig
appconfig - Sets application configuration class name
autocommit
autocommit — Enable or disable automatic transaction commit mode
batch
batch — Start or execute a batch of SQL statements
brief
brief — Enable terse output mode
close
close — Close the active connection
columns
columns — Display columns of a table
commandhandler
commandhandler — Adds command handlers with commands
commit
commit — Commit the current transaction
connect
connect — Connect to a database
dbinfo
dbinfo — Provide information about the current database
describe
describe — Describe a table
dropall
dropall — Drop all tables in the database or in the schema
exportedkeys
exportedkeys — List exported foreign keys for a database

go
go — Change to a different active connection
help
help — Display help information
history
history — Display the command history
-d Print timestamps for each event (default for !history)
-f Print full time date stamps in the US format
-E Print full time date stamps in the European format
-i Print full time date stamps in ISO8601 format
-n Suppresses command numbers
-r Reverses the order of the commands
importedkeys
importedkeys — List imported foreign keys for a database
indexes
indexes — Display indexes for a table
isolation
isolation — Set the transaction isolation mode for the active connection
list
list — Display all active connections
metadata
metadata — Invoke arbitrary metadata commands
outputformat
outputformat — Change the method for displaying SQL results
primarykeys
primarykeys — Display the primary key columns for a table
procedures
procedures — List stored procedures
prompthandler
prompthandler - Set custom prompt handler class name
properties
properties — Connect to the database defined in the specified properties file.
quit
quit — Exit SQLLine
reconnect
reconnect — Reconnect to the current database
record
record — Begin recording all output from SQL commands
rehash
rehash — Obtain a list of all tables and columns from the database
rollback
rollback — Roll back the current transaction
run
run — Execute a command script
save
save — Save the current preferences
schemas
schemas — List all the schemas in the database
scan
scan — Scan class path for JDBC drivers
script
script — Save executed commands to a file
set
set — Set a preference
sql
sql — Execute a SQL against a database
tables
tables — List all the tables in the database
verbose
verbose — Enable verbose output
all

Name

all — Execute SQL against all active connections

Synopsis

!all {SQL statement}

Description

Execute the specified SQL against all the current connections.

Example of "all" command

1: jdbc:mysql://localhost/mydb> !list

2 active connections:
#0 open jdbc:oracle:thin:@localhost:1521:mydb
#1 open jdbc:mysql://localhost/mydb

1: jdbc:mysql://localhost/mydb> !all DELETE FROM COMPANY;

Executing SQL against: jdbc:oracle:thin:@localhost:1521:mydb
4 rows affected (0.004 seconds)

Executing SQL against: jdbc:mysql://localhost/mydb
1 row affected (3.187 seconds)
autocommit

Name

autocommit — Enable or disable automatic transaction commit mode

Synopsis

!autocommit {on/off}

Description

Set autocommit mode on or off. When autocommit is on, every individual SQL statement will be committed after it is issued. Otherwise, the commit command will need to be issued in order to commit any changes to the database.

Example of "autocommit" command

1: jdbc:mysql://localhost/mydb> !autocommit off

Autocommit status: false

1: jdbc:mysql://localhost/mydb>

batch

Name

batch — Start or execute a batch of SQL statements

Synopsis

!batch

Description

Start or execute a batch of SQL statements. The first time the statement is issued, subsequent SQL statements will be deferred until the batch command is issued again. Databases can frequently optimize multiple batched statements (e.g., for bulk data loading) in order to speed up execution.

Example of "batch" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !batch

Batching SQL statements. Run "batch" again to execute the batch.

0: jdbc:oracle:thin:@localhost:1521:mydb> INSERT INTO MYTABLE (ID) VALUES (1);
0: jdbc:oracle:thin:@localhost:1521:mydb> INSERT INTO MYTABLE (ID) VALUES (2);
0: jdbc:oracle:thin:@localhost:1521:mydb> INSERT INTO MYTABLE (ID) VALUES (3);
0: jdbc:oracle:thin:@localhost:1521:mydb> INSERT INTO MYTABLE (ID) VALUES (4);
0: jdbc:oracle:thin:@localhost:1521:mydb> !batch

Running batched SQL statements...

<table>
<thead>
<tr>
<th>COUNT</th>
<th>STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>INSERT INTO MYTABLE (ID) VALUES (1)</td>
</tr>
<tr>
<td>1</td>
<td>INSERT INTO MYTABLE (ID) VALUES (2)</td>
</tr>
<tr>
<td>1</td>
<td>INSERT INTO MYTABLE (ID) VALUES (3)</td>
</tr>
<tr>
<td>1</td>
<td>INSERT INTO MYTABLE (ID) VALUES (4)</td>
</tr>
</tbody>
</table>

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM MYTABLE;

+-----+
| ID  |
+-----+
| 1   |
| 2   |
| 3   |
| 4   |
+-----+
4 rows selected (0.012 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> brief

Name

brief — Enable terse output mode

Synopsis

!brief

Description

Set verbose mode off. This will prevent printing of stack traces when errors occur.

Example of "brief" command

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM BOGUS_TABLE;

Error: ORA-00942: table or view does not exist (state=42000,code=942)
java.sql.SQLException: ORA-00942: table or view does not exist
   at oracle.jdbc.dbaccess.DBError.throwSQLException(DBError.java:180)
   at oracle.jdbc.ttc7.TTLoer.processError(TTLoer.java:208)
verbose: off

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM BOGUS_TABLE;
Error: ORA-00942: table or view does not exist (state=42000, code=942)

Name

close — Close the active connection

Synopsis

!close

Description

Close the current connection and remote it from the connection list.

Example of "close" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !list

1 active connection:
#0 open    jdbc:oracle:thin:@localhost:1521:mydb

0: jdbc:oracle:thin:@localhost:1521:mydb> !close

Closing: oracle.jdbc.driver.OracleConnection

sqline> !list
No active connections

sqlline>
columns

Name
columns — Display columns of a table

Synopsis

!columns \{table name\}

Description

List the columns in the specified table.

Example of "columns" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !columns EMPLOYEE

+-------------+--------------+------------+------------+
| TABLE_NAME  | COLUMN_NAME  | DATA_TYPE  | TYPE_NAME  |
+-------------+--------------+------------+------------+
| EMPLOYEE    | ID           | 3          | NUMBER     |
| EMPLOYEE    | FIRST_NAME   | 12         | VARCHAR2   |
| EMPLOYEE    | LAST_NAME    | 12         | VARCHAR2   |
+-------------+--------------+------------+------------+

0: jdbc:oracle:thin:@localhost:1521:mydb>

commandhandler

Name

commandhandler — Adds command handlers with commands

Synopsis

!commandhandler \{command handler\} \{command handler\}*

Description

Adds command handlers with commands. Command handler class must have a constructor accepting Sqlline and implement sqlline.CommandHandler
Example of "commandhandler" command

sqline> '!hello
Unknown command: hello
sqline> '!commandhandler sqline.extensions.HelloWorldCommandHandler
sqline> '!hello
HELLO WORLD

commit

Name

commit — Commit the current transaction

Synopsis

'!commit

Description

Commit the current transaction, if autocommit is off.

Example of "commit" command

0: jdbc:oracle:thin:@localhost:1521:mydb> '!commit

Commit complete (0.661 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM EMPLOYEE;

+--------------+-------------+------------+
| ID           | FIRST_NAME  | LAST_NAME  |
+--------------+-------------+------------+
| 1            | Bob         | Smith      |
+--------------+-------------+------------+
1 row selected (0.309 seconds)
connect

Name

connect — Connect to a database

Synopsis

`!connect [-p property value]* {url} [username password] [driver class]`

Description

Connect to a database. Quotes could be used in case of having spaces

Example of "connect" command

sqline> !connect jdbc:oracle:thin:@localhost:1521:mydb scott tiger oracle.jdbc.driver.OracleDriver

Connected to: Oracle (version Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production)
Driver: Oracle JDBC driver (version 9.0.1.0.0)
Autocommit status: true

0: jdbc:oracle:thin:@localhost:1521:mydb>

dbinfo

Name

dbinfo — Provide information about the current database

Synopsis

`!dbinfo`

Description

List metadata information about the current connection.

Example of "dbinfo" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !dbinfo

allProceduresAreCallable false
allTablesAreSelectable false
dataDefinitionCausesTransactionCommit true
dataDefinitionIgnoredInTransactions false
doesMaxRowSizeIncludeBlobs          true
getCatalogSeparator
getCatalogTerm
getDatabaseProductName              Oracle
getDatabaseProductVersion           Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production

With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production
getDefaultTransactionIsolation      2
getDriverMajorVersion               9
getDriverMinorVersion               0
getDriverName                       Oracle JDBC driver
getDriverVersion                    9.0.1.0.0
getExtraNameCharacters              $#
getIdentifierQuoteString            "
getMaxBinaryLiteralLength           1000
getMaxCatalogNameLength             0
getMaxCharLiteralLength             2000
getMaxColumnNameLength              30
getMaxColumnsInGroupBy              0
getMaxColumnsInIndex                32
getMaxColumnsInOrderBy              0
getMaxColumnsInSelect               0
getMaxColumnsInTable                1000
getMaxConnections                   0
getMaxCursorNameLength              0
getMaxIndexLength                   0
getMaxProcedureNameLength           30
getMaxRowSize                       2000
getMaxSchemaNameLength              30
getMaxStatementLength               65535
getMaxStatements                    0
getMaxTableNameLength               30
getMaxTablesInSelect                0
getMaxUserNameLength                30
getNumericFunctions                 ABS, CEIL, COS, COSH, EXP, FLOOR, LN, LOG, MOD, POWER, ROUND, SIGN, SIN, SINH, SQRT, TAN, TANH, TRUNC, AVG, COUNT, GLB, LUB, MAX, MIN, STDDEV, SUM, VARIANCE

getProcedureTerm                    procedure
getSchemaTerm                       schema
getSearchStringEscape               //
getSQLKeywords                      ACCESS, ADD, ALTER, AUDIT, CLUSTER, COLUMN, COMMENT, COMPRESS, CONNECT, DATE, DROP, EXCLUSIVE, FILE, IDENTIFIED, IMMEDIATE, INCREMENT, INDEX, INITIAL, INTERSECT,
LEVEL, LOCK, LONG, MAXEXTENTS, MINUS, MODE, NOAUDIT, NOCOMRESS, NOWAIT, NUMBER, OFFLINE, ONLINE, PCTFREE, PRIOR,

all_PL_SQL_reserved_words

getStringFunctions
CHR, INITCAP, LOWER, LPAD, LTRIM, NLS_INITCAP, NLS_LOWER, NLS_UPPER, REPLACE, RPAD, RTRIM, SOUNDEX, SUBSTR, SUBSTRB, TRANSLATE, UPPER, ASCII, INSTR, INSTRB, LENGTH, LENGTHB, NLSSORT, CHARTOROWID, CONVERT, HEXTORAW, RAWTOHEX, ROWIDTOCHAR, TO_CHAR, TO_DATE, TO_LABEL, TO_MULTI_BYTE, TO_NUMBER, TO_SINGLE_BYTE

getSystemFunctions
DUMP, GREATEST, GREATEST_LB, LEAST, LEAST_UB, NVL, UID, USER, USERENV, VSIZE

getTimeDateFunctions
ADD_MONTHS, LAST_DAY, MONTHS_BETWEEN, NEW_TIME, NEXT_DAY, ROUND, SYSDATE, TRUNC

getURL
jdbc:oracle:thin:@localhost:1521:mydb

getUserName
MYDB

isCatalogAtStart
false

isReadOnly
false

nullPlusNonNullIsNull
true

nullsAreSortedAtEnd
false

nullsAreSortedAtStart
false

nullsAreSortedHigh
false

nullsAreSortedLow
true

storesLowerCaseIdentifiers
false

storesLowerCaseQuotedIdentifiers
false

storesUpperCaseIdentifiers
true

storesUpperCaseQuotedIdentifiers
false

supportsAlterTableWithAddColumn
true

supportsAlterTableWithDropColumn
false

supportsANSI92EntryLevelSQL
true

supportsANSI92FullSQL
false

supportsANSI92IntermediateSQL
false

supportsBatchUpdates
true

supportsCatalogsInDataManipulation
false
<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>supportsCatalogsInIndexDefinitions</td>
<td>false</td>
</tr>
<tr>
<td>supportsCatalogsInPrivilegeDefinitions</td>
<td>false</td>
</tr>
<tr>
<td>supportsCatalogsInProcedureCalls</td>
<td>false</td>
</tr>
<tr>
<td>supportsCatalogsInTableDefinitions</td>
<td>false</td>
</tr>
<tr>
<td>supportsColumnAliasing</td>
<td>true</td>
</tr>
<tr>
<td>supportsConvert</td>
<td>true</td>
</tr>
<tr>
<td>supportsCoreSQLGrammar</td>
<td>true</td>
</tr>
<tr>
<td>supportsCorrelatedSubqueries</td>
<td>true</td>
</tr>
<tr>
<td>supportsDataDefinitionAndDataManipulationTransactions</td>
<td>true</td>
</tr>
<tr>
<td>supportsData Manipulation Transactions Only</td>
<td>true</td>
</tr>
<tr>
<td>supportsDifferentTableCorrelationNames</td>
<td>true</td>
</tr>
<tr>
<td>supportsExpressionsInOrderBy</td>
<td>true</td>
</tr>
<tr>
<td>supportsExtendedSQLGrammar</td>
<td>true</td>
</tr>
<tr>
<td>supportsFullOuterJoins</td>
<td>true</td>
</tr>
<tr>
<td>supportsGroupBy</td>
<td>true</td>
</tr>
<tr>
<td>supportsGroupByBeyondSelect</td>
<td>true</td>
</tr>
<tr>
<td>supportsGroupByUnrelated</td>
<td>true</td>
</tr>
<tr>
<td>supportsIntegrityEnhancementFacility</td>
<td>true</td>
</tr>
<tr>
<td>supportsLikeEscapeClause</td>
<td>true</td>
</tr>
<tr>
<td>supportsLimitedOuterJoins</td>
<td>true</td>
</tr>
<tr>
<td>supportsMinimumSQLGrammar</td>
<td>true</td>
</tr>
<tr>
<td>supportsMixedCaseIdentifiers</td>
<td>false</td>
</tr>
<tr>
<td>supportsMixedCaseQuotedIdentifiers</td>
<td>true</td>
</tr>
<tr>
<td>supportsMultipleResultSets</td>
<td>false</td>
</tr>
<tr>
<td>supportsMultipleTransactions</td>
<td>true</td>
</tr>
<tr>
<td>supportsNonNullColumns</td>
<td>true</td>
</tr>
<tr>
<td>supportsOpenCursorsAcrossCommit</td>
<td>false</td>
</tr>
<tr>
<td>supportsOpenCursorsAcrossRollback</td>
<td>false</td>
</tr>
<tr>
<td>supportsOpenStatementsAcrossCommit</td>
<td>false</td>
</tr>
<tr>
<td>supportsOpenStatementsAcrossRollback</td>
<td>false</td>
</tr>
<tr>
<td>supportsOrderByUnrelated</td>
<td>true</td>
</tr>
<tr>
<td>supportsOuterJoins</td>
<td>true</td>
</tr>
<tr>
<td>supportsPositionedDelete</td>
<td>true</td>
</tr>
<tr>
<td>supportsPositionedUpdate</td>
<td>true</td>
</tr>
<tr>
<td>supportsSchemasInDataManipulation</td>
<td>true</td>
</tr>
<tr>
<td>supportsSchemasInIndexDefinitions</td>
<td>true</td>
</tr>
<tr>
<td>supportsSchemasInPrivilegeDefinitions</td>
<td>true</td>
</tr>
<tr>
<td>supportsSchemasInProcedureCalls</td>
<td>true</td>
</tr>
<tr>
<td>supportsSchemasInTableDefinitions</td>
<td>true</td>
</tr>
<tr>
<td>supportsSelectForUpdate</td>
<td>true</td>
</tr>
<tr>
<td>supportsStoredProcedures</td>
<td>true</td>
</tr>
<tr>
<td>supportsSubqueriesInComparisons</td>
<td>true</td>
</tr>
<tr>
<td>supportsSubqueriesInExists</td>
<td>true</td>
</tr>
<tr>
<td>supportsSubqueriesInIns</td>
<td>true</td>
</tr>
<tr>
<td>supportsSubqueriesInQuantifieds</td>
<td>true</td>
</tr>
<tr>
<td>supportsTableCorrelationNames</td>
<td>true</td>
</tr>
<tr>
<td>supportsTransactions</td>
<td>true</td>
</tr>
<tr>
<td>supportsUnion</td>
<td>true</td>
</tr>
</tbody>
</table>
supportsUnionAll: true
usesLocalFilePerTable: false
usesLocalFiles: false

0: jdbc:oracle:thin:@localhost:1521:mydb>
describe

Name

describe — Describe a table

Synopsis

!describe [table name]

Description

Synonym for columns (if an argument is specified) or tables (if no argument is specified).

Example of "describe" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !describe EMPLOYEE

+--------------+------------+-------------+-----------+------------+
| TABLE_SCHEM  | TABLE_NAME | COLUMN_NAME | DATA_TYPE | TYPE_NAME  |
+--------------+------------+-------------+-----------+------------+
| MYDB         | EMPLOYEE   | ID          | 3         | NUMBER     |
| MYDB         | EMPLOYEE   | FIRST_NAME  | 12        | VARCHAR2   |
| MYDB         | EMPLOYEE   | LAST_NAME   | 12        | VARCHAR2   |
+--------------+------------+-------------+-----------+------------+

0: jdbc:oracle:thin:@localhost:1521:mydb>
dropall

Name

dropall — Drop all tables in the database or in the schema

Synopsis

!dropall [schema_pattern]

Description

Drop every table in the database.
Example of "dropall" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !dropall

Really drop every table in the database? (y/n) y

1/5 DROP TABLE COMPANY;
2/5 DROP TABLE EMPLOYEE;
3/5 DROP TABLE PROJECTS;
4/5 DROP TABLE EMPLOYEE_PROJECTS;
5/5 DROP TABLE COMPANYEMPLOYEES;

0: jdbc:oracle:thin:@localhost:1521:mydb>

Warning on using "dropall"

Warning

This command will destroy all data in the database (subject to access restrictions). Use with extreme caution.

exportedkeys

Name

exportedkeys — List exported foreign keys for a database

Synopsis

!exportedkeys {table name}

Description

List all the foreign keys that are exported by the specified table.

Example of "exportedkeys" command

0: jdbc:hsqldb:db-hypersonic> !exportedkeys COMPANY

+----------------+---------------+----------------+-------------+
| PKTABLE_SCHEM  | PKTABLE_NAME  | PKCOLUMN_NAME  | FKTABLE_CAT |
|----------------+---------------+----------------+-------------+
|                | COMPANY       | COMPANY_ID     |             |
+----------------+---------------+----------------+-------------+

0: jdbc:hsqldb:db-hypersonic>

go
Name

go — Change to a different active connection

Synopsis

!go {connection number}

Description

Switch to the specified connection.

Example of "go" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !list

2 active connections:
#0 open  jdbc:oracle:thin:@localhost:1521:mydb
#1 open  jdbc:mysql://localhost/mydb

0: jdbc:oracle:thin:@localhost:1521:mydb> !go 1

1: jdbc:mysql://localhost/mydb> !go 0

0: jdbc:oracle:thin:@localhost:1521:mydb>

help

Name

help — Display help information

Synopsis

!help

Description

Display information about allowed commands.

Example of "help" command

sqlline> !help
!all Execute the specified SQL against all the current connections
!appconfig Set custom application configuration class name
!autocommit Set autocommit mode on or off
!batch    Start or execute a batch of statements
!brief    Set verbose mode off
!call     Execute a callable statement
!close    Close the current connection to the database
!closeall Close all current open connections
!columns  List all the columns for the specified table
!commandhandler Add a command handler
!commit   Commit the current transaction (if autocommit is off)
!connect  Open a new connection to the database
!dbinfo   Give metadata information about the database
!describe Describe a table
!dropall  Drop all tables in the current database
!exportedkeys List all the exported keys for the specified table
!go       Select the current connection
!help     Print a summary of command usage
!history  Display the command history
          -d Print timestamps for each event (default for !history)
          -f Print full time date stamps in the US format
          -E Print full time date stamps in the European format
          -i Print full time date stamps in ISO8601 format
          -n Suppresses command numbers
          -r Reverses the order of the commands
!importedkeys List all the imported keys for the specified table
!indexes  List all the indexes for the specified table
!isolation Set the transaction isolation for this connection
!list     List the current connections
!manual   Display the SQLLine manual
!metadata Obtain metadata information
!nativesql Show the native SQL for the specified statement
!nickname Create a friendly name for the connection (updates command prompt)
!outputformat Set the output format for displaying results (table, vertical, csv, tsv, xmlattrs, xmlelements, json, ansiconsole)
!primarykeys List all the primary keys for the specified table
!procedures List all the procedures
!prompthandler Set custom prompt handler class name
!properties Connect to the database specified in the properties file(s)
!quit     Exits the program
!reconnect Reconnect to the database
!record   Record all output to the specified file
!rehash   Fetch table and column names for command completion
!rollback Roll back the current transaction (if autocommit is off)
!run      Run a script from the specified file
!save     Save the current variables and aliases
!schemas  List all the schemas in the database
!scan     Scan for installed JDBC drivers
!script   Start saving a script to a file
!set      Set a sqlline variable
!sql                Execute a SQL command
!tables             List all the tables in the database
!typeinfo           Display the type map for the current connection
!verbose            Set verbose mode on

Variables:

Variable        Value      Description
=============== ========== ==================================================
autoCommit      true/false Enable/disable automatic transaction commit
autoSave        true/false Automatically save preferences
color           true/false Control whether color is used for display
colorScheme     chester/dark/dracula/light/obsidian/solarized/vs2010 Syntax highlight schema
confirm         true/false Prompts for confirmation before running commands specified in confirmPattern
confirmPattern  pattern    Pattern commands to issue prompt before execution
csvDelimiter    String     Delimiter in csv outputFormat
csvQuoteCharacter char     Quote character in csv outputFormat
dateFormat      pattern    Format dates using SimpleDateFormat pattern
escapeOutput    true/false Escape control symbols in output
fastConnect     true/false Skip building table/column list for tab-completion
force           true/false Continue running script even after errors
headerInterval  integer    The interval between which headers are displayed
historyFile     path       File in which to save command history. Default is $HOME/.sqlline/history (UNIX, Linux, Mac OS), $HOME/sqlline/history (Windows)
incremental     true/false Do not receive all rows from server before printing the first row; uses fewer resources, especially for long-running queries, but column widths may be incorrect
incrementalBufferRows integer Enter incremental mode if a query returns more than "incrementalBufferRows".
isolation       LEVEL      Set transaction isolation level
maxColumnWidth  integer    The maximum width to use when displaying columns
maxHeight       integer    The maximum height of the terminal
maxWidth        integer    The maximum width of the terminal
maxHistoryFileRows integer The maximum number of history rows to store in history file
maxHistoryRows  integer    The maximum number of history rows to store in memory
mode            emacs/vi   The editing mode
nullValue       String     Use String in place of NULL values
numberFormat    pattern    Format numbers using DecimalFormat pattern
outputFormat    table/vertical/csv/tsv/xmlattrs/xmllelements/json/ansiconsole Format mode for result display
prompt          pattern    Format prompt
propertiesFile  path     File from which SQLLine reads properties on startup; default is $HOME/.sqlline/sqlline.properties (UNIX, Linux, macOS), $HOME/sqlline/sqlline.properties (Windows)
rightPrompt      pattern  Format right prompt
rowLimit         integer   Maximum number of rows returned from a query; zero means no limit
showCompletionDesc true/false Display help for completions
showElapsedTime  true/false Display execution time when verbose
showHeader       true/false Show column names in query results
showNestedErrs   true/false Display nested errors
showWarnings     true/false Display connection warnings
silent           true/false Be more silent
strictJdbc       true/false Use strict JDBC
timeFormat       pattern  Format times using SimpleDateFormat pattern
timeout          integer   Query timeout in seconds; less than zero means no timeout
timestampFormat  pattern  Format timestamps using SimpleDateFormat pattern
trimScripts      true/false Remove trailing spaces from lines read from script files
useLineContinuation true/false Use line continuation
verbose          true/false Show verbose error messages and debug info

Key-strokes:
alt-b            Backward word
alt-c            Capitalize word
alt-d            Kill word
alt-f            Forward word
alt-h            Next color scheme
alt-l            Lowercase word
alt-n            History search forward
alt-p            History search backward
alt-t            Transpose words
alt-u            Uppercase word
ctrl-a           To the beginning of line
ctrl-b           Backward char
ctrl-d           Delete char
ctrl-e           To the end of line
ctrl-f           Forward char
ctrl-h           Backward delete char
ctrl-i           Complete
ctrl-j           Enter
ctrl-k           Kill the line
ctrl-m           Enter
ctrl-n           Down line or history
ctrl-p           Up line or history
ctrl-r           History incremental search backward
Open Source Used In DNAC Platform Arica

ctrl-s History incremental search forward
ctrl-t Transpose chars
ctrl-u Kill the whole line
ctrl-w Backward kill the line

Comments, bug reports, and patches go to ???
sqlline>

history

Name

history — Display the command history
   -d Print timestamps for each event (default for !history)
   -f Print full time date stamps in the US format
   -E Print full time date stamps in the European format
   -i Print full time date stamps in ISO8601 format
   -n Suppresses command numbers
   -r Reverses the order of the commands

Synopsis

!history

Description

Display the command history.

Example of "history" command

sqlline> !history

1. `outputformat xml elements
2. SELECT * FROM PCPOINT;
3. `outputformat vertical
4. SELECT * FROM PCPOINT;
5. `tables

sqlline>

importedkeys

Name

importedkeys — List imported foreign keys for a database

Synopsis
!importedkeys {table name}

Description

List all the foreign keys that are imported by the specified table.

Example of "importedkeys" command

0: jdbc:hsqldb:db-hypersonic> !importedkeys EMPLOYEE

+----------------+---------------+----------------+-------------+
| PKTABLE_SCHEM   | PKTABLE_NAME  | PKCOLUMN_NAME   | FKTABLE_CAT |
+----------------+---------------+----------------+-------------+
|                | COMPANY       | COMPANY_ID      |             |
+----------------+---------------+----------------+-------------+
0: jdbc:hsqldb:db-hypersonic>

indexes

Name

indexes — Display indexes for a table

Synopsis

!indexes {table name}

Description

List all the indexes on the specified table name.

Example of "indexes" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !indexes EMPLOYEE

+-------------+-------------+-----------------+-------+--------------+
| TABLE_NAME  | NON_UNIQUE  | INDEX_NAME      | TYPE  | COLUMN_NAME  |
+-------------+-------------+-----------------+-------+--------------+
| EMPLOYEE    | 0           |                 | 0     |              |
| EMPLOYEE    | 0           | SYS_C003115849  | 1     | ID           |
| EMPLOYEE    | 1           | FIRST_NAME      | 1     | FIRST_NAME   |
| EMPLOYEE    | 1           | LAST_NAME       | 1     | LAST_NAME    |
+-------------+-------------+-----------------+-------+--------------+
0: jdbc:oracle:thin:@localhost:1521:mydb>
isolation

Name

isolation — Set the transaction isolation mode for the active connection

Synopsis

!isolation {isolation level}

Description

Set the isolation level for the current transaction. For a description of the different isolation levels, see https://java.sun.com/j2se/1.3/docs/api/java/sql/Connection.html#field_detail. If the specified isolation level is not supported or DEFAULT is specified then there will be used level returned by java.sql.DatabaseMetaData#getDefaultTransactionIsolation

Example of "isolation" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !isolation TRANSACTION_READ_UNCOMMITTED

Error: READ_COMMITTED and SERIALIZABLE are the only valid transaction levels (state=,code=17030)

0: jdbc:oracle:thin:@localhost:1521:mydb> !isolation TRANSACTION_SERIALIZABLE

Transaction isolation: TRANSACTION_SERIALIZABLE

0: jdbc:oracle:thin:@localhost:1521:mydb>

list

Name

list — Display all active connections

Synopsis

!list

Description

List all the current connections.

Example of "list" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !list

2 active connections:
metadata

Name

metadata — Invoke arbitrary metadata commands

Synopsis

`!metadata {methodname} {params...}`

Description

Execute an arbitrary metadata method against the current connection. refpurpose are separated by spaces. Use `""` for a blank String, and null for a null parameter. For information on available metadata methods, see https://java.sun.com/j2se/1.3/docs/api/java/sql/DatabaseMetaData.html.

Example of "metadata" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !metadata supportsSelectForUpdate

true

0: jdbc:oracle:thin:@localhost:1521:mydb> !metadata getTypeInfo

+------------+------------+-------------+-----------------+-----------------+
| TYPE_NAME  | DATA_TYPE  |  PRECISION  | LITERAL_PREFIX  | LITERAL_SUFFIX  |
+------------+------------+-------------+-----------------+-----------------+
| NUMBER     | -7         | 1           |                 |                 |
| NUMBER     | -6         | 3           |                 |                 |
| NUMBER     | -5         | 38          |                 |                 |
| LONG RAW   | -4         | 2147483647  | '               | '               |
| RAW        | -3         | 2000        | '               | '               |
| LONG       | -1         | 2147483647  | '               | '               |
| CHAR       | 1          | 2000        | '               | '               |
| NUMBER     | 2          | 38          |                 |                 |
| NUMBER     | 4          | 10          |                 |                 |
| NUMBER     | 5          | 5           |                 |                 |
| FLOAT      | 6          | 63          |                 |                 |
| REAL       | 7          | 63          |                 |                 |
| VARCHAR2   | 12         | 4000        | '               | '               |
| DATE       | 93         | 7           |                 |                 |
| STRUCT     | 2002       | 0           | '               | '               |
| ARRAY      | 2003       | 0           | '               | '               |
outputformat

Name

outputformat — Change the method for displaying SQL results

Synopsis

!outputformat {format name}

Description

Set the mode for displaying results from statements. This is useful for saving output from SQL statements to either a visually pleasing format or an easily parseable format.

In the case of csv or table outputformat there could be specified additional parameters like csv delimiter, csv quote character, max column width.

Such settings will change corresponding properties' values. If the parameters not specified then properties are left as is.

Example of table output formatting

0: jdbc:oracle:thin:@localhost:1521:mydb> !outputformat table
0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;

+----------+-------+
| COMPANY_ID | NAME   |
|-----------+-------|
| 1         | Apple  |
| 2         | Sun    |
| 3         | IBM    |
| 4 | Microsoft |
+--------------------------+
4 rows selected (0.012 seconds)

Example of vertical output formatting

0: jdbc:oracle:thin:@localhost:1521:mydb> !outputformat vertical
0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;
COMPANY_ID  1
NAME        Apple
COMPANY_ID  2
NAME        Sun
COMPANY_ID  3
NAME        IBM
COMPANY_ID  4
NAME        Microsoft
4 rows selected (0.011 seconds)

Example of CSV output formatting

0: jdbc:calcite:model=target/test-classes/mod> !set outputFormat csv ### @
0: jdbc:calcite:model=target/test-classes/mod> SELECT * FROM COMPANY;
@COMPANY_ID@###@NAME@
@1@###@Apple@
@2@###@Sun@
@3@###@IBM@
@4@###@Microsoft@
4 rows selected (0.014 seconds)

Example of TSV output formatting

0: jdbc:oracle:thin:@localhost:1521:mydb> !outputformat tsv
0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;
'COMPANY_ID'    'NAME'
'1'     'Apple'
'2'     'Sun'
'3'     'IBM'
'4'     'Microsoft'
4 rows selected (0.013 seconds)

Example of XML attribute output formatting

0: jdbc:oracle:thin:@localhost:1521:mydb> !outputformat xmlattr
0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;
Example of XML element output formatting

```
0: jdbc:oracle:thin:@localhost:1521:mydb> !outputformat xmlelements
0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;
<resultset>
<result>
    <COMPANY_ID>1</COMPANY_ID>
    <NAME>Apple</NAME>
</result>
<result>
    <COMPANY_ID>2</COMPANY_ID>
    <NAME>Sun</NAME>
</result>
<result>
    <COMPANY_ID>3</COMPANY_ID>
    <NAME>IBM</NAME>
</result>
<result>
    <COMPANY_ID>4</COMPANY_ID>
    <NAME>Microsoft</NAME>
</result>
</resultset>
4 rows selected (0.02 seconds)
0: jdbc:oracle:thin:@localhost:1521:mydb>
```

Example of JSON output formatting

```
0: jdbc:calcite:model=target/test-classes/mod> !outputformat json
0: jdbc:calcite:model=target/test-classes/mod> SELECT * FROM COMPANY;
{"resultset":[
{"COMPANY_ID":1,"NAME":"Apple"},
{"COMPANY_ID":2,"NAME":"Sun"},
{"COMPANY_ID":3,"NAME":"IBM"},
{"COMPANY_ID":4,"NAME":"Microsoft"}
]
4 rows selected (0.03 seconds)
0: jdbc:calcite:model=target/test-classes/mod>
```

Example of AnsiConsole output formatting

```
0: jdbc:calcite:model=target/test-classes/mod> !outputformat json
0: jdbc:calcite:model=target/test-classes/mod> SELECT * FROM COMPANY;
["resultset":[
["COMPANY_ID":1,"NAME":"Apple"],
["COMPANY_ID":2,"NAME":"Sun"],
["COMPANY_ID":3,"NAME":"IBM"],
["COMPANY_ID":4,"NAME":"Microsoft"]
]]
4 rows selected (0.03 seconds)
0: jdbc:calcite:model=target/test-classes/mod>
```
0: jdbc:calcite:model=target/test-classes/mod> !outputformat ansiconsole
0: jdbc:calcite:model=target/test-classes/mod> SELECT * FROM COMPANY;
COMPANY_ID  NAME
1           Apple
2           Sun
3           IBM
4           Microsoft
4 rows selected (0.03 seconds)
0: jdbc:calcite:model=target/test-classes/mod>

primarykeys

Name

primarykeys — Display the primary key columns for a table

Synopsis

!primarykeys {table name}

Description

List all the primary keys for the specified table.

Example of "primarykeys" command

0: jdbc:hsqldb:db-hypersonic> !primarykeys COMPANY

+---------------+-------------+--------------+----------+----------+
| TABLE_SCHEM   | TABLE_NAME  | COLUMN_NAME  | KEY_SEQ  | PK_N     |
|               | COMPANY     | COMPANY_ID   | 1        | SYS_PK_C |
+---------------+-------------+--------------+----------+----------+
0: jdbc:hsqldb:db-hypersonic>

procedures

Name

procedures — List stored procedures

Synopsis

!procedures [procedure name pattern]

Description
Display a list of all the stored procedures that are exposed by the current database.

Example of "procedures" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !procedures %JDBC%

+----------------+------------------+------------------+
| PROCEDURE_CAT  | PROCEDURE_SCHEM  |  PROCEDURE_NAME  |
+----------------+------------------+------------------+
| WK_ADM         | WKSYS            | GET_JDBC_DRIVER  |
| WK_ADM         | WKSYS            | SET_JDBC_DRIVER  |
+----------------+------------------+------------------+

0: jdbc:oracle:thin:@localhost:1521:mydb>

properties

Name

properties — Connect to the database defined in the specified properties file.

Synopsis

!properties {file}

Description

Connect to the database defined in the specified properties file. The properties file is expected to have the following values:

url: the database URL to which to connect

driver: the driver class that handles the URL. If unspecified, SQILine with automatically scan the CLASSPATH for an appropriate driver

user: the username to use to connect to the database. If unspecified, SQILine will prompt for it.

password: the password to use to connect to the database. If unspecified, SQILine will prompt for it.

The key names in the properties file can also use JDO semantics for the properties. These properties are "javax.jdo.option.ConnectionURL", "javax.jdo.option.ConnectionDriverName", "javax.jdo.option.ConnectionUserName", and "javax.jdo.option.ConnectionPassword". Furthermore, any properties that end with "ConnectionURL", "ConnectionDriverName", "ConnectionUserName", and "ConnectionPassword" will be used for the connection.

Example of "properties" command

sqlline> !properties test.properties
Connecting to jdbc:mysql://localhost/mydb

Enter password for jdbc:mysql://localhost/mydb: *****

Connected to: MySQL (version 3.23.52-log)
Driver: MySQL-AB JDBC Driver (version 3.0.8-stable ( $Date: 2008/12/10 $, $Revision: #3 $ ))
Autocommit status: true
Transaction isolation: TRANSACTION_READ_COMMITTED

0: jdbc:mysql://localhost/mydb>

quit

Name

quit — Exit SQLLine

Synopsis

!quit

Description

Exit SQLLine.

Example of "reconnect" command

0: jdbc:hsqldb:db-hypersonic> !quit

Closing: org.hsqldb.jdbcConnection

reconnect

Name

reconnect — Reconnect to the current database

Synopsis

!reconnect

Description

Reconnect to the current database.

Example of "reconnect" command
0: jdbc:oracle:thin:@localhost:1521:mydb> !reconnect

Reconnecting to "jdbc:oracle:thin:@localhost:1521:mydb"...

Closing: oracle.jdbc.driver.OracleConnection@4428d3

Connected to: Oracle (version Oracle9i Enterprise Edition Release 9.2.0.1.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.1.0 - Production)
Driver: Oracle JDBC driver (version 9.0.1.0.0)
Autocommit status: true

0: jdbc:oracle:thin:@localhost:1521:mydb>

record

Name

record — Begin recording all output from SQL commands

Synopsis

!record {file name}

Description

Save all output from the session to the specified file.

Example of "record" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !record /tmp/mysession.out

Saving all output to "/tmp/mysession.out". Enter "record" with no arguments to stop it.

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;
+-------------+------------+
<table>
<thead>
<tr>
<th>COMPANY_ID</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>IBM</td>
</tr>
<tr>
<td>4</td>
<td>Microsoft</td>
</tr>
<tr>
<td>1</td>
<td>Apple</td>
</tr>
<tr>
<td>2</td>
<td>Sun</td>
</tr>
</tbody>
</table>
+-------------+------------+
4 rows selected (0.011 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> !record

Recording stopped.
rehash

Name

rehash — Obtain a list of all tables and columns from the database

Synopsis

!rehash

Description

Get a list of all tables and columns from the database in order to include them in the list for tab-completion of SQL statements. This is done automatically on connect when the fastconnect option is enabled.

Example of "rehash" command

0: jdbc:hsqldb:db-hypersonic> !rehash

Building list of tables and columns for tab-completion (set fastconnect to true to skip)...
7/7 (100%) Done
Done

0: jdbc:hsqldb:db-hypersonic>

Warning on using "rehash"

Caution

This operation can be extremely slow for some databases.

rollback

Name

rollback — Roll back the current transaction

Synopsis

!rollback

Description

Rollback the current transaction, if autocommit is off.
Example of "rollback" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !autocommit off

Autocommit status: false

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;

+-------------+------------+
| COMPANY_ID  |    NAME    |
+-------------+------------+
| 3           | IBM        |
| 4           | Microsoft  |
| 1           | Apple      |
| 2           | Sun        |
+-------------+------------+
4 rows selected (0.011 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> DELETE FROM COMPANY;

4 rows affected (0.004 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;

+-------------+-------+
| COMPANY_ID  | NAME  |
+-------------+-------+
|             |       |
+-------------+-------+
No rows selected (0.01 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> !rollback

Rollback complete (0.016 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM COMPANY;

+-------------+------------+
| COMPANY_ID  |    NAME    |
+-------------+------------+
| 3           | IBM        |
| 4           | Microsoft  |
| 1           | Apple      |
| 2           | Sun        |
+-------------+------------+
4 rows selected (0.01 seconds)

0: jdbc:oracle:thin:@localhost:1521:mydb>
run

Name

run — Execute a command script

Synopsis

!run {file name}

Description

Run the individual commands specified in the file name. The file should consist of individual SQL statements or SQLLine commands. Lines beginning with "#" are interpreted as comments and ignored. If any errors occur while running the script, the script will be aborted, unless the force preference is set to true.

Example of "run" command

0: jdbc:hsqldb:db-hypersonic> !run example.sql
1/11 CREATE TABLE COMPANY (COMPANY_ID INT, NAME VARCHAR(255));
No rows affected (0.001 seconds)
2/11 INSERT INTO COMPANY VALUES (1, 'Apple');
1 row affected (0 seconds)
3/11 INSERT INTO COMPANY VALUES (2, 'Sun');
1 row affected (0 seconds)
4/11 INSERT INTO COMPANY VALUES (3, 'IBM');
1 row affected (0.001 seconds)
5/11 INSERT INTO COMPANY VALUES (4, 'Microsoft');
1 row affected (0.015 seconds)
6/11 CREATE TABLE EMPLOYEE (ID INT, FIRST_NAME VARCHAR(255), LAST_NAME VARCHAR(255), COMPANY INT);
No rows affected (0.004 seconds)
7/11 CREATE INDEX FIRST_NAME ON EMPLOYEE(FIRST_NAME);
No rows affected (0 seconds)
8/11 CREATE INDEX LAST_NAME ON EMPLOYEE(LAST_NAME);
No rows affected (0.001 seconds)
9/11 ALTER TABLE EMPLOYEE ADD CONSTRAINT EMP_COMPANY FOREIGN KEY (COMPANY) REFERENCES COMPANY (COMPANY_ID);
No rows affected (0 seconds)
10/11 INSERT INTO EMPLOYEE (ID, FIRST_NAME, LAST_NAME, COMPANY) VALUES (234, 'William', 'Gates', 4);
1 row affected (0.001 seconds)
11/11 SELECT * FROM COMPANY,EMPLOYEE WHERE EMPLOYEE.COMPANY = COMPANY.COMPANY_ID;
+------------+-----------+--------------+-------------+------------+---------+
| COMPANY_ID |    NAME   | ID           | FIRST_NAME  | LAST_NAME  | COMPANY |
+------------+-----------+--------------+-------------+------------+---------+
Name

save — Save the current preferences

Synopsis

!save

Description

Save the current preferences.

Example of "save" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !save

Saving preferences to: /Users/mprudhom/.sqlline/sqlline.properties

0: jdbc:oracle:thin:@localhost:1521:mydb>

Name

scan — Scan class path for JDBC drivers

Synopsis

!scan

Description

Scans all the jar files in the CLASSPATH for any JDBC drivers, and prints the class names of the drivers.

Example of "scan" command

sqlline> !scan

35 driver classes found
Compliant Version Driver Class
no  4.0  COM.cloudscape.core.JDBCDriver
no  1.0  COM.cloudscape.core.RmiJdbcDriver
yes  7.1  COM.ibm.db2.jdbc.net.DB2Driver
yes  3.2  com.ddtek.jdbc.db2.DB2Driver
yes  3.2  com.ddtek.jdbc.informix.InformixDriver
yes  3.2  com.ddtek.jdbc.oracle.OracleDriver
yes  3.2  com.ddtek.jdbc.sqlserver.SQLServerDriver
yes  3.2  com.ddtek.jdbc.sybase.SybaseDriver
yes  1.0  com.ibm.db2.jcc.DB2Driver
yes  2.21 com.informix.jdbc.IfxDriver
no  0.2  com.internetcds.jdbc.tds.Driver
no  0.2  com.internetcds.jdbc.tds.SybaseDriver
yes  2.2715  com.jnetdirect.jsql.JSQLDriver
yes  3.2  com.merant.datadirect.jdbc.db2.DB2Driver
yes  3.2  com.merant.datadirect.jdbc.informix.InformixDriver
yes  3.2  com.merant.datadirect.jdbc.oracle.OracleDriver
yes  3.2  com.merant.datadirect.jdbc.sqlserver.SQLServerDriver
yes  3.2  com.merant.datadirect.jdbc.sybase.SybaseDriver
yes  2.2  com.microsoft.jdbc.sqlserver.SQLServerDriver
no  3.0  com.mysql.jdbc.Driver
no  3.0  com.mysql.jdbc.NonRegisteringDriver
yes  4.4  com.pointbase.jdbc.jdbcDriver
yes  4.4  com.pointbase.jdbc.jdbcEmbeddedDriver
yes  4.4  com.pointbase.jdbc.jdbcUniversalDriver
no  5.5  com.sybase.jdbc2.jdbc.SybDriver
yes  1.0  in.co.daffodil.db.jdbc.DaffodilDBDriver
no  2.0  interbase.interclient.Driver
no  0.5  net.sourceforge.jtds.jdbc.Driver
yes  1.0  oracle.jdbc.OracleDriver
yes  1.0  oracle.jdbc.driver.OracleDriver
no  3.26  org.enhydra.instantdb.jdbc.idbDriver
no  3.0  org.gjt.mm.mysql.Driver
no  1.7  org.hsqldb.jdbcDriver
no  7.3  org.postgresql.Driver
yes  0.9  org.sourceforge.jdbc.JXDBConDriver

sqline>

schemas

Name

schemas — List all the schemas in the database

Synopsis

!schemas
Description

List all the schemas in the database of the current connection.

Example of "schemas" command

0: jdbc:postgresql:thin:@localhost:5432:mydb> !schemas

+-------------------+---------------+
| table_schem       | table_catalog |
+-------------------+---------------+
| information_schema |               |
| pg_catalog        |               |
| public            |               |
+-------------------+---------------+

0: jdbc:postgresql:thin:@localhost:5432:mydb>

script

Name

script — Save executed commands to a file

Synopsis

!script {filename}

Description

Start saving all commands entered to the specified file. Once scripting on, it can be finished by entering the script command again with no arguments. Saved scripts can be re-run using the run command.

Example of "script" command

0: jdbc:hsqldb:db-hypersonic> !script /tmp/mysession.script

Saving command script to "/tmp/mysession.script". Enter "script" with no arguments to stop it.

0: jdbc:hsqldb:db-hypersonic> !autocommit off

Autocommit status: false

0: jdbc:hsqldb:db-hypersonic> SELECT * FROM COMPANY;

+-------------+------------+
| COMPANY_ID  |    NAME    |
+-------------+------------+
| 1 | Apple      | |
| 2 | Sun        | |
| 3 | IBM        | |
| 4 | Microsoft  | |

+-----------------------+

4 rows selected (0.001 seconds)

0: jdbc:hsqldb:db-hypersonic> UPDATE COMPANY SET NAME='Apple, Inc.' WHERE NAME='Apple';

1 row affected (0.005 seconds)

0: jdbc:hsqldb:db-hypersonic> 'commit

Commit complete (0 seconds)

0: jdbc:hsqldb:db-hypersonic> !script

Script closed. Enter "run /tmp/mysession.script" to replay it.

0: jdbc:hsqldb:db-hypersonic>

set

Name

set — Set a preference

Synopsis

!set {preference} {value}

Description

Set the specified preference to the specified value. If autosave is on, then this will cause preferences to be immediately saved. Otherwise, changed setting can be saved for future SQLLine session eith the save command.

Example of "set" command

0: jdbc:hsqldb:db-hypersonic> !set color on

sql

Name

sql — Execute a SQL against a database

Synopsis
!sql {statement}

Description

Issue the specified SQL statement. This is equivalent to entering the statement by itself from the command line.

Example of "sql" command

0: jdbc:hsqldb:db-hypersonic> !sql DELETE FROM COMPANY;

4 rows affected (0.005 seconds)

0: jdbc:hsqldb:db-hypersonic>

tables

Name

tables — List all the tables in the database

Synopsis

!tables

Description

List all the tables in the database of the current connection.

Example of "tables" command

0: jdbc:oracle:thin:@localhost:1521:mydb> !tables

+------------+--------------+---------------------------------+-------------+
| TABLE_CAT  | TABLE_SCHEM  |           TABLE_NAME            | TABLE_TYPE  |
+------------+--------------+---------------------------------+-------------+
|            | MYDB         | COMPANY                          | TABLE       |
|            | MYDB         | EMPLOYEE                         | TABLE       |
+------------+--------------+---------------------------------+-------------+

0: jdbc:oracle:thin:@localhost:1521:mydb>

verbose

Name

verbose — Enable verbose output

Synopsis
!verbose

Description

Enable verbose mode, which causes stack traces to be printed when errors occur, and enabled outputting of debug information.

Example of "verbose" command

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM BOGUS_TABLE;

Error: ORA-00942: table or view does not exist (state=42000,code=942)

0: jdbc:oracle:thin:@localhost:1521:mydb> !verbose

verbose: on

0: jdbc:oracle:thin:@localhost:1521:mydb> SELECT * FROM BOGUS_TABLE;

Error: ORA-00942: table or view does not exist (state=42000,code=942)
java.sql.SQLException: ORA-00942: table or view does not exist
    at oracle.jdbc.dbaccess.DBError.throwSqlException(DBError.java:180)
    at oracle.jdbc.ttc7.TTIoer.processError(TTIoer.java:208)
    at oracle.jdbc.ttc7.Oall7.receive(Oall7.java:543)
    at oracle.jdbc.ttc7.TTC7Protocol.doOall7(TTC7Protocol.java:1451)
    at oracle.jdbc.ttc7.TTC7Protocol.parseExecuteDescribe(TTC7Protocol.java:651)
    at oracle.jdbc.driver.OracleStatement.doExecuteQuery(OracleStatement.java:2110)
    at oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:2324)
    at oracle.jdbc.driver.OracleStatement.execute(OracleStatement.java:909)
    at sqlline.SqlLine$Commands.sql(SqlLine.java:3183)
    at sqlline.SqlLine.dispatch(SqlLine.java:771)
    at sqlline.SqlLine.begin(SqlLine.java:634)
    at sqlline.SqlLine.main(SqlLine.java:332)

0: jdbc:oracle:thin:@localhost:1521:mydb>

Parameter Reference

autocommit
autosave
color
colorScheme
confirm
confirmPattern
csvDelimiter
csvQuoteCharacter
dateformat
escapeOutput
fastconnect
force
headerinterval
historyfile
incremental
isolation
maxcolumnwidth
maxHistoryFileRows
maxHistoryRows
maxwidth
mode
nullValue
numberformat
outputformat
prompt
rightprompt
rowlimit
showheader
shownestederrs
showtime
showwarnings
silent
strictJdbc
timeformat
timestampformat
trimscripts
uselinecontinuation
verbose
autocommit

If true, then new connections will have autocommit set, otherwise, transactions will need to be explicitly committed or rolled back. Defaults to true. To change the autocommit status for a connection that is already open and active, use the autocommit command instead.

autosave

When set to true, any changes to preferences using the set command will cause the preferences to be saved. Otherwise, preferences will need to be explicitly saved using the save command. Defaults to false.

color

If true, then output to the terminal will use color for a more pleasing visual experience. Requires that the terminal support ANSI control codes (most do). Defaults to false.

colorScheme
If chester/dark/dracula/light/obsidian/solarized/vs2010, then this scheme will be used for command/sql syntax highlighting. If default then there is no syntax highlighting.

confirm

If true, then user will be prompted for confirmation if commands matching with confirmPattern are executed.

confirmPattern

If default with confirm = true, user will be prompted for DROP and DELETE commands, otherwise, for the commands as matched in the pattern.

csvDelimiter

Sets the delimiter in csv outputFormat. Setting to default causes usage of "," as a delimiter.

csvQuoteCharacter

Sets the quote character in csv outputFormat. Setting to default causes usage of "/" as a delimiter.

dateformat

The format for how date values are displayed. Setting to default causes date values to be fetched and rendered via ResultSet.getString. Any other setting results in fetch via ResultSet.getObject and rendering via java.text.SimpleDateFormat. For example, the setting "YYYY_MM_dd" yields values like "1970_01_01".

escapeOutput

When set to true, control symbols in output will be escaped. Otherwise they will be printed as is. Default is false.

fastconnect

When false, any new connection will cause SQLLine to access information about the available tables and columns in order to provide them as candidates for tab-completion. This can be a very slow operation for some databases, do by default it is off. Table and column information can always be explicitly retrieved using the rehash command.

force

When set to false, any failures during the execution of the run or record commands will cause execution to abort, preventing the execution of subsequent commands. When set to true, errors will be displayed but otherwise ignored. Defaults to false.

headerinterval

The interval between which the column headers will be redisplayed when using the "table" outputformat. Defaults to 100.

historyfile
The file to which SQLLine will save a record of all the commands issued. Defaults to HOME/sqlline/history on
Windows, and HOME/.sqlline/history on all other platforms.

incremental

When set to false, the entire result set is fetched and buffered before being displayed, yielding optimal display
column sizing. When set to true, result rows are displayed immediately as they are fetched, yielding lower latency
and memory usage at the price of extra display column padding. Defaults to false.

incrementalBufferRows

When incremental set to false and the entire result set has more rows than value of incrementalBufferRows, sqlline
will enter incremental mode. In that case, result rows are displayed by batches of size incrementalBufferRows. If
value is negative, sqlline never enters incremental mode. Defaults to 1000.

isolation

The default transaction isolation that will be used for new connections. To change the isolation level of the currently
active connection, use the isolation command instead.

maxcolumnwidth

The maximum column width to display for each column before truncating data when using the "table" outputformat.
Defaults to 15.

maxHistoryFileRows

The maximum number of history rows to store in history file. Defaults to 10000
(org.jline.reader.impl.history.DefaultHistory.DEFAULT_HISTORY_FILE_SIZE).

maxHistoryRows

The maximum number of history rows to store in memory. Defaults to 500
(org.jline.reader.impl.history.DefaultHistory.DEFAULT_HISTORY_SIZE).

maxwidth

The maximum width to display before truncating data when using the "table" outputformat. Defaults to attempting
to query the terminal for the current width, falls back to 80.

mode

The editing mode to use in sqlline. Defaults to "emacs".

nullvalue

The value which will be used instead of null values. Setting to default causes 'null' for numbers/dates/times and
empty strings for others. For example, the setting *** yields null values like ‘***’

numberformat

The format for how numeric values are displayed. Setting to default causes numeric values to be fetched and rendered via ResultSet.getString. Any other setting results in fetch via ResultSet.getObject and rendering via java.text.DecimalFormat. For example, the setting "0.###E0" yields scientific notation with up to three fractional digits, values like "6.022E23".

outputformat

The format for how results are displayed. For details, see the information on the outputformat command.

prompt

The tcsh-like format for how prompt is displayed. For example, the setting ‘%\033[1;33msqlline\033[m>’ yields yellow sqlline and normal angle bracket. Defaults to ‘sqlline>’. If for the specific database connection nickname is set then nickname will be used.

rightprompt

The tcsh-like format for how right prompt is displayed. The same patterns are applied as for prompt but it does not care if nickname is set or not. Defaults to “”.

promptscript

A JavaScript program that is evaluated to generate a prompt. If specified, prompt is ignored.

The following variables are pre-defined for use in the JavaScript: connectionIndex, currentSchema, databaseProductName, url, userName. For example:

```
sqlline> !set promptscript 'databaseProductName.toLowerCase() + ":" + connectionIndex + "> "'
```

rowlimit

The maximum number of rows to fetch per query. Defaults to 0, which is interpreted as fetching all rows.

showheader

If true, display the names of the columns when displaying results. Defaults to true.

shownestederrs

If true, display any nested errors that are reported on the connection after issuing any database commands. Defaults to false, meaning only the topmost error is displayed.

showtime
If true, display execution time for database commands when in verbose mode. Defaults to true.

showwarnings

If true, display any warnings that are reported on the connection after issuing any database commands. Defaults to true.

silent

If true, then reduce the amount of informational messages displayed. Useful for redirecting a sqlline command to a file for later parsing. Defaults to false.

strictJdbc

If false, then use default values defined by java.sql.DatabaseMetaData in case an implementation of this interface fails with e.g. "Method not supported", otherwise it also fails. Defaults to false.

timeformat

The format for how time values are displayed. Setting to default causes time values to be fetched and rendered via ResultSet.getString. Any other setting results in fetch via ResultSet.getObject and rendering via java.text.SimpleDateFormat. For example, the setting "HH:mm:ss" yields values like "14:12:11".

timestampformat

The format for how timestamp values are displayed. Setting to default causes timestamp values to be fetched and rendered via ResultSet.getString. Any other setting results in fetch via ResultSet.getObject and rendering via java.text.SimpleDateFormat. For example, the setting "dd/MM/YYYY'T'HH:mm:ss" yields values like "01/01/1970T12:32:12".

trimscripts

If true, then trim leading and trailing whitespace from lines as they are processed in scripts; otherwise, preserve whitespace. Defaults to true.

uselinecontinuation

Use line continuation if true. Defaults to true.

verbose

If true, then print out the entire java stack trace whenever an error occurs, as well as displaying debugging information. Defaults to false.

JDBC Driver Support

Project Information
1.2822 clean-stack 2.2.0

1.2822.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2823 tunnel-agent 0.6.0 0.6.0

1.2823.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2824 global-prefix 1.0.2 1.0.2
1.2824.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2825 is-number 3.0.0
1.2825.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2826 jsreport-exceljs 3.7.1 3.7.1

1.2826.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2019 Guyon Roche

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2827 @babel/plugin-transform-destructuring 7.16.0
1.2827.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2828 buffer-more-ints 0.0.2 0.0.2

1.2828.1 Available under license:
Copyright (c) 2012 David Wragg. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2829 object.omit 2.0.1
1.2829.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014-2016, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"). to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2830 protostuff 1.7.1
1.2830.1 Available under license:

Thanks to
===========

Michael Donohue

* J2ME module

Alexander Dvorkovyy

* Deployment to maven central

Ivan Prisyazhniy

* `proto_extender` (compiler/codegen)
* `java.bean_model` (compiler/codegen)

Igor Scherbak
* `proto_extender` (compiler/codegen)

John Koepi

* `ProtoToProtoCompiler`
* `JavaBeanModelCompiler`

Patch contributors
==================

J.D. Zamfirescu

* `protostuff-parser` - supplied code to parse protobuf "syntax2" header. (Issue 6)

Philippe Laflamme

* `protostuff-parser` - supplied code to parse protobuf extensions and allow generated code to access its metadata. (Issue 25)

Joseph Gentle

* `protostuff-model` - fixed bug in the reflection compiler with nested protobuf types. (Issue 26)

Tim Underwood

* `protostuff-api` - ArrayIndexOutOfBoundsException in StringSerializer.writeUTF8 (Issue 27)

nordlig.ulv

* `protostuff-compiler` - Native methods for repeated fields (Issue 35)
* `protostuff-compiler` - Merge method for json protobuf classes (Issue 64)

avaskys

* `protostuff-parser`, `protostuff-compiler` - cannot import more than one .proto file from the same package (Issue 38)

sergei.s.ivanov

* `protostuff-compiler` Use FQCN everywhere in classes generated by java_v2protoc_schema (Issue 70). Maven plugin to attach generated source directory to the project compile source roots (Issue 69).

Thomas Kock

* `protostuff-uberjar` - a bundle created primarily for osgi.

thierry.monney
* 'protostuff-me' - Preverification fails (Issue 92)

Leo Romanoff

* 'protostuff-runtime' - user definable mapping: FQNC's --> integer, for serialized stream size-reduction (Issue 96)

Brice Jaglin

* 'protostuff-runtime' - annotations on pojo's fields to have explicit control of field numbers.

baetz.alexander

* 'protostuff-compiler' - Issue 128 patch

* 'protostuff-maven-plugin' - Issue 128 patch

Max Lunin

* 'protostuff-core' - Issue 138

Ryszard Winiewski

* 'protostuff-runtime' - Issue 140 patch

lzh0379

* 'protostuff-runtime' - Issue 141 patch

eezjlee

* 'protostuff-runtime-md' - Issue 156 patch

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protostuff
Copyright 2009 David Yu dyuproject@gmail.com

protobuf is copyright Google inc unless otherwise noted.
It is licensed under the BSD license.
Jackson-core-asl is copyright FasterXml unless otherwise noted.
It is licensed under the apache 2.0 license.

Antrr is copyright Terence Parr unless otherwise noted.
It is licensed under the BSD license.

Stringtemplate is copyright Terence Parr unless otherwise noted.
It is licensed under the BSD license.

Velocity is licensed under the apache 2.0 license.

B64Code.java is copyright Mort Bay Consulting Pty Ltd unless otherwise noted.
It is licensed under the apache 2.0 license.

Jarjar is copyright Google inc unless otherwise noted.
It is licensed under the apache 2.0 license.

Guava is copyright Google inc unless otherwise noted.
It is licensed under the apache 2.0 license.

1.2831 aws-sign2 0.7.0 0.7.0
1.2831.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source
code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2832 silent-spawn 0.4.0 0.4.0
1.2832.1 Available under license:
MIT License

Copyright (c) 2017 Jan Blaha

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2833 is-data-descriptor 0.1.4
1.2833.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2834 hbase-metrics 2.4.2

1.2834.1 Available under license:

Apache HBase - Metrics Implementation
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2835 xmldom 0.1.27 0.1.27
1.2835.1 Available under license:

You can choose any one of those:

The MIT License (MIT):

link:http://opensource.org/licenses/MIT

LGPL:

http://www.gnu.org/licenses/lgpl.html

1.2836 has-flag 3.0.0
1.2836.1 Available under license:
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2837 profiler 1.0.2

1.2838 core-js 1.2.7 1.2.7
1.2838.1 Available under license:
Copyright (c) 2015 Denis Pushkarev

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2839 dnsjava 2.1.7

1.2839.1 Available under license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2840 parquet-jackson 1.8.1

1.2840.1 Available under license:

Apache Parquet Jackson
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained in the "NOTICE" text file.
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be...
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product depends on Jackson and includes it in this binary artifact. Jackson is a high-performance JSON processor.

Copyright: 2007-2015 Tatu Saloranta and other contributors
Home page: http://jackson.codehaus.org/
License: http://www.apache.org/licenses/LICENSE-2.0.txt

1.2841 bouncy-castle 1.57
1.2841.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2842 hsts 1.0.0 1.0.0
1.2842.1 Available under license :
The MIT License (MIT)

Copyright (c) 2014-2015 Evan Hahn, Adam Baldwin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2843 de.huxhorn.sulky.formatting 0.9.17
1.2843.1 Available under license :
GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.
As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time
a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.2844 jaxb-api 2.3.1

1.2844.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or
portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the
terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION.
THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as “Participant”) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ?252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48
C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

---------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This
alternative is allowed only for noncommercial distribution and only if you received the program in object code or
executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable
work, complete source code means all the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is normally distributed (in either source or
binary form) with the major components (compiler, kernel, and so on) of the operating system on which the
executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering
equivalent access to copy the source code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License.
Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically
terminate your rights under this License. However, parties who have received copies, or rights, from you under this
License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you
permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do
not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program),
you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or
modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives
a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions.
You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited
to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the
conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as
to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a
consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-
free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only
way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the
section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest
validity of any such claims; this section has the sole purpose of protecting the integrity of the free software
distribution system, which is implemented by public license practices. Many people have made generous
contributions to the wide range of software distributed through that system in reliance on consistent application of
that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted
interfaces, the original copyright holder who places the Program under this License may add an explicit
geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this
License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time
to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new
problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License
which applies to it and "any later version", you have the option of following the terms and conditions either of that
version or of any later version published by the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are
different, write to the author to ask for permission. For software which is copyrighted by the Free Software
Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be
guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE
PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM
"AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY
COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL,
SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY
TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER
PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.2845 should-equal 2.0.0 2.0.0

1.2845.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2846 **picomatch 2.2.2 2.2.2**

1.2846.1 Available under license:

The MIT License (MIT)

Copyright (c) 2017-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2847 **has-color 0.1.7 0.1.7**

1.2847.1 Available under license:

The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2848 lodash-merge 4.6.2 4.6.2

1.2848.1 Available under license:

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====
Files located in the node_modules and vendor directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2849 @babel/plugin-transform-classes
7.16.0
1.2849.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2850 groovyjsr 3.0.8
1.2850.1 Available under license:
No license file was found, but licenses were detected in source scan.

.baomidou*/
* The initial contribution was derived from the reference implementation
* developed by Sun in consultation with the Groovy community. The reference
* implementation had the following license header:
* 
* Copyright 2006 Sun Microsystems, Inc. All rights reserved.
* Use is subject to license terms.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Sun Microsystems nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Subsequent modifications by the Groovy community have been done under the Apache License v2:

Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/
No license file was found, but licenses were detected in source scan.

# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements. See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License. You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

org.codehaus.groovy.jsr223.GroovyScriptEngineFactory

No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
 */
1.2851 ajv-keywords 3.4.1 3.4.1

1.2851.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2852 redux 3.7.2 3.7.2

1.2852.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-present Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2853 puppeteer 2.0.0 2.0.0

1.2853.1 Available under license :
MIT License

Copyright (c) Microsoft Corporation. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2854 pinkie-promise 2.0.1 2.0.1

1.2854.1 Available under license :
The MIT License (MIT)

Copyright (c) Vsevolod Strukchinsky <floatdrop@gmail.com> (github.com/floatdrop)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2855 ajv-keywords 3.5.2 3.5.2

1.2855.1 Available under license:
The MIT License (MIT)

Copyright (c) 2016 Evgeny Poberezkin

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2856 akka 2.5.32
1.2856.1 Available under license:

Copyright (c) 2002-2006, Marc Prud'hommeaux<br>Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "["]
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
This software includes the following components from the Apache Software Foundation, also
released under the Apache Software License Version 2.0:

commons-cli-1.1.jar
jakarta-oro-2.0.8.jar
log4j-1.2.15.jar

The following components are covered under different licenses:

jline-0.9.94.jar (LICENSE-JLINE)
One-Jar (LICENSE-ONEJAR)
Java/Tcl v1.3.3 (LICENSE-TCLJAVA,-2,-3,-4)
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

---------------

Licenses for dependency projects can be found here:
[http://akka.io/docs/akka/snapshot/project/licenses.html]

akka-protobuf contains the sources of Google protobuf 2.5.0 runtime support, moved into the source package `akka.protobuf` so as to avoid version conflicts. For license information see COPYING.protobuf

The following license terms apply to the TJC compiler source and test files located in the src/tjc, src/tests/tjc, and tests/tjc directories.

2005 Advanced Micro Devices, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that existing copyright notices are retained in all copies, this notice is included verbatim in any distributions, and the terms and conditions hererin are met.

Use of the this software manifests acceptance of the terms of this
license by performance.

The name of Advanced Micro Devices, Inc. may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY ADVANCED MICRO DEVICES, INC. "AS IS" AND ANY EXPRESS, IMPLIED OR STATUTORY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT, OR THOSE ARISING FROM CUSTOM OF TRADE OR COURSE OF USAGE ARE DISCLAIMED.

IN NO EVENT SHALL ADVANCED MICRO DEVICES, INC. BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED, AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE ITS DOCUMENTATION OR ANY DERIVATIVES THEREOF, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. BY USING THIS SOFTWARE WITHOUT CHARGE, YOU ACCEPT THIS ALLOCATION OF RISK. THIS DISCLAIMER OF LIABILITY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. ADVANCED MICRO DEVICES, INC. HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS OF THIS SOFTWARE.

In the redistribution and use of this software, each party shall at all times comply with all applicable governmental laws, statutes, ordinances, rules, regulations, orders, and other requirements, including without limitation such governmental requirements applicable to environmental protection, health, safety, wages, hours, equal employment opportunity, nondiscrimination, working conditions, import or export control, and transportation. Without limiting the foregoing, each party shall adhere to the U.S. Export Administration Regulations (EAR), currently found at 15 C.F.R. Sections 730 through 744, and, unless properly authorized by the U.S. Government, shall not (1) export, re-export or release restricted technology, software, or source code to a national of a country in Country Groups D:1 or E:1, or (2) export to Country Groups D:1 or E:1 the direct product of such technology or software, if such foreign produced direct product is subject to national security controls as identified on the Commerce Control List (currently found in Supplement 1 to Section 774 of EAR). These export requirements shall survive any expiration or termination of this agreement.

# Licenses

## Akka License

```
This software is licensed under the Apache 2 license, quoted below.
```

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## Akka Committer License Agreement

All committers have signed this [CLA](http://www.lightbend.com/contribute/current-cla). It can be [signed online](http://www.lightbend.com/contribute/cla).

## Licenses for Dependency Libraries

Each dependency and its license can be seen in the project build file (the comment on the side of each dependency):

@extref[AkkaBuild.scala](github:project/AkkaBuild.scala#L1054)

Portions of Jacl and Tcl Blend are Copyright (c) 1997-1999 The Regents of the University of California. All rights reserved.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
The following license terms apply to the Itcl source and test files located in the src/itcl and tests/itcl directories.

This software is copyrighted by Cadence Design Systems, Inc., and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

-----------------------------------------------------------------------
Following is the original agreement for the Tcl/Tk software from Sun Microsystems.

-----------------------------------------------------------------------
This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Apache Commons CLI
Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

SUN MICROSYSTEMS, INC. THROUGH ITS SUN MICROSYSTEMS LABORATORIES
DIVISION ("SUN") WILL LICENSE THIS SOFTWARE AND THE ACCOMPANYING
DOCUMENTATION TO YOU (a "Licensee") ONLY ON YOUR ACCEPTANCE OF ALL
THE TERMS SET FORTH BELOW.

Sun grants Licensee a non-exclusive, royalty-free right to download, install, compile, use, copy and distribute the Software, modify or otherwise create derivative works from the Software (each, a "Modification") and distribute any Modification in source code and/or binary code form to its customers with a license agreement containing these terms and noting that the Software has been modified. The Software is copyrighted by Sun and other third parties and Licensee shall retain and reproduce all copyright and other notices presently on the Software. As between Sun and Licensee, Sun is the sole owner of all rights in and to the Software other than the limited rights granted to Licensee herein; Licensee will own its Modifications, expressly subject to Sun's continuing ownership of the Software. Licensee will, at its expense, defend and indemnify Sun and its licensors from and against any third party claims, including costs and reasonable attorneys' fees, and be wholly responsible for any
liabilities arising out of or related to Licensee's development, use
or distribution of the Software or Modifications. Any distribution of
the Software and Modifications must comply with all applicable United
States export control laws.

THE SOFTWARE IS BEING PROVIDED TO LICENSEE "AS IS" AND ALL EXPRESS OR
IMPLIED CONDITIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT,
ARE DISCLAIMED. IN NO EVENT WILL SUN BE LIABLE HEREUNDER FOR ANY
DIRECT DAMAGES OR ANY INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES OF ANY KIND.
The jmxsh library has been obtained from https://code.google.com/p/jmxsh/ under the Apache License 2.0.
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
- Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
Apache log4j
Copyright 2007 The Apache Software Foundation
This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

/* ====================================================================
   * The Apache Software License, Version 1.1
   *
   * Copyright (c) 2000-2002 The Apache Software Foundation. All rights
   * reserved.
   *
   * Redistributions and use in source and binary forms, with or without
   * modification, are permitted provided that the following conditions
   * are met:
   *
   * 1. Redistributions of source code must retain the above copyright
   *    notice, this list of conditions and the following disclaimer.
   *
   * 2. Redistributions in binary form must reproduce the above copyright
   *    notice, this list of conditions and the following disclaimer in
   *    the documentation and/or other materials provided with the
   *    distribution.
   *
   * 3. The end-user documentation included with the redistribution,
   *    if any, must include the following acknowledgment:
   *    "This product includes software developed by the
   *     Apache Software Foundation (http://www.apache.org/)."
   *    Alternately, this acknowledgment may appear in the software itself,
   *    if and wherever such third-party acknowledgments normally appear.
   *
   * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
   *    must not be used to endorse or promote products derived from this
   *    software without prior written permission. For written
   *    permission, please contact apache@apache.org.
   *
   * 5. Products derived from this software may not be called "Apache"
   *    or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
   *    name, without prior written permission of the Apache Software Foundation.
   *
   * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED
   * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
   * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
   * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
   * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
   * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
   * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
   * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
   * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
   * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
   * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
   * SUCH DAMAGE.
   */
This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

* One-JAR (http://www.simontuffs.com/one-jar). Copyright (c) 2004-2007, P. Simon Tuffs (simon@simontuffs.com). All rights reserved.

* Redistributions and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of P. Simon Tuffs, nor the names of any contributors, nor the name One-JAR may be used to endorse or promote products derived from this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Including this file inside the built One-JAR file conforms with these terms.

1.2857 libjpeg-turbo 1.4.2-0ubuntu3.3

1.2857.1 Available under license:

libjpeg-turbo Licenses

libjpeg-turbo is covered by three compatible BSD-style open source licenses:
- The IJG (Independent JPEG Group) License, which is listed in [README.ijg](README.ijg)

This license applies to the libjpeg API library and associated programs (any code inherited from libjpeg, and any modifications to that code.)

- The Modified (3-clause) BSD License, which is listed below

This license covers the TurboJPEG API library and associated programs, as well as the build system.

- The [zlib License](https://opensource.org/licenses/Zlib)

This license is a subset of the other two, and it covers the libjpeg-turbo SIMD extensions.

**Complying with the libjpeg-turbo Licenses**

This section provides a roll-up of the libjpeg-turbo licensing terms, to the best of our understanding.

1. If you are distributing a modified version of the libjpeg-turbo source, then:

1. You cannot alter or remove any existing copyright or license notices from the source.

   **Origin**
   - Clause 1 of the IJG License
   - Clause 1 of the Modified BSD License
   - Clauses 1 and 3 of the zlib License

2. You must add your own copyright notice to the header of each source file you modified, so others can tell that you modified that file (if there is not an existing copyright header in that file, then you can simply add a notice stating that you modified the file.)

   **Origin**
   - Clause 1 of the IJG License
   - Clause 2 of the zlib License

3. You must include the IJG README file, and you must not alter any of the copyright or license text in that file.

   **Origin**
2. If you are distributing only libjpeg-turbo binaries without the source, or if you are distributing an application that statically links with libjpeg-turbo, then:

1. Your product documentation must include a message stating:

   This software is based in part on the work of the Independent JPEG Group.

   **Origin**
   - Clause 2 of the IJG license

2. If your binary distribution includes or uses the TurboJPEG API, then your product documentation must include the text of the Modified BSD License (see below.)

   **Origin**
   - Clause 2 of the Modified BSD License

3. You cannot use the name of the IJG or The libjpeg-turbo Project or the contributors thereof in advertising, publicity, etc.

   **Origin**
   - IJG License
   - Clause 3 of the Modified BSD License

4. The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be free of defects, nor do we accept any liability for undesirable consequences resulting from your use of the software.

   **Origin**
   - IJG License
   - Modified BSD License
   - zlib License

The Modified (3-clause) BSD License
====================================

Copyright (C)2009-2020 D. R. Commander. All Rights Reserved.
Copyright (C)2015 Viktor Szathmry. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, 
  this list of conditions and the following disclaimer in the documentation 
  and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its 
  contributors may be used to endorse or promote products derived from this 
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS", 
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE 
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR 
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF 
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS 
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN 
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) 
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE 
POSSIBILITY OF SUCH DAMAGE.

Why Three Licenses?
===================

The zlib License could have been used instead of the Modified (3-clause) BSD 
License, and since the IJG License effectively subsumes the distribution 
conditions of the zlib License, this would have effectively placed 
libjpeg-turbo binary distributions under the IJG License. However, the IJG 
License specifically refers to the Independent JPEG Group and does not extend 
attrtribution and endorsement protections to other entities. Thus, it was 
desirable to choose a license that granted us the same protections for new code 
that were granted to the IJG for code derived from their software.

1.2858 union-value 1.0.1 1.0.1
1.2858.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy 
of this software and associated documentation files (the "Software"), to deal 
in the Software without restriction, including without limitation the rights 
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the Software is 
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2859 redux-thunk 2.3.0 2.3.0

1.2859.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015-present Dan Abramov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2860 cheerio 0.22.0 0.22.0

1.2860.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2861 regexp-quote 0.0.0 0.0.0

1.2861.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2862 map-cache 0.2.2 0.2.2

1.2862.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2863 swagger-schema-official 2.0.0-bab6bed 2.0.0-bab6bed

1.2863.1 Available under license:

The MIT License (MIT)

Copyright (c) 2012 Mike Ihbe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2864 netty-codec 4.1.59.Final

1.2864.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/DefaultHeadersImpl.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/CharSequenceValueConverter.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/ValueConverter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 */
* under the License.
*/
/**
* Written by Robert Harder and released to the public domain, as explained at
* https://creativecommons.org/licenses/publicdomain
*/
/**
* Enumeration of supported Base64 dialects.
* <p>
* The internal lookup tables in this class has been derived from
* <a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain
* Base64 Encoder/Decoder</a>.
*/

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/base64/Base64Dialect.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/META-
INF/maven/io.netty/netty-codec/pom.xml
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/marshalling/ThreadLocalMarshallerProvider.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/UnsupportedMessageTypeException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/serialization/ObjectEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/protobuf/package-info.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/LineBasedFrameDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/marshalling/MarshallingDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/SnappyFrameDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/DecoderException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/Delimiters.java
Open Source Used In DNAC Platform Arica 13993

* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/LengthFieldPrepender.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/string/StringDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/MessageToByteEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/ByteToMessageCodec.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/FixedLengthFrameDecoder.java
Open Source Used In DNAC Platform Arica 13994

* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/ByteToMessageDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/CorruptedFrameException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/base64/Base64Encoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/bytes/ByteArrayDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/serialization/ClassLoaderClassResolver.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/marshalling/MarshallerProvider.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/CodecException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/base64/Base64Decoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/serialization/ClassResolvers.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/CodecException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/PrematureChannelClosureException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/EncoderException.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/MessageAggregator.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/serialization/CachingClassResolver.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/base64/package-info.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/MessageToMessageEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/serialization/ClassResolver.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/DelimiterBasedFrameDecoder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/DateFormatter.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-jar/io/netty/handler/codec/MessageFormatter.java
jar/io/netty/handler/codec/CodecOutputList.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/string/LineSeparator.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/string/LineEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/compression/CompressionUtil.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/compression/ByteBufChecksum.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/DatagramPacketEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/DatagramPacketDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32FrameDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32LengthFieldPrepender.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufDecoderNano.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufEncoderNano.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/HeadersUtils.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32FrameDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufVarint32LengthFieldPrepender.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtocolDetectionResult.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufDecoderNano.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
jar/io/netty/handler/codec/protobuf/ProtobufEncoderNano.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/**
 * A decoder that splits the received {@link ByteBuf}s dynamically by the
 * value of the length field in the message. It is particularly useful when you
 * decode a binary message which has an integer header field that represents the
 * length of the message body or the whole message.
 * 
 * {@link LengthFieldBasedFrameDecoder} has many configuration parameters so
 * that it can decode any message with a length field, which is often seen in
 * proprietary client-server protocols. Here are some example that will give
 * you the basic idea on which option does what.
 *
 * <h3>2 bytes length field at offset 0, do not strip header</h3>
 *
 * The value of the length field in this example is <tt>12 (0x0C)</tt> which
 * represents the length of "HELLO, WORLD". By default, the decoder assumes
 * that the length field represents the number of the bytes that follows the
 * length field. Therefore, it can be decoded with the simplistic parameter
 * combination.
 *
 * BEFORE DECODE (14 bytes) AFTER DECODE (14 bytes)
 * +--------+----------------+      +--------+----------------+
 * | Length | Actual Content |----->| Length | Actual Content |
Because we can get the length of the content by calling

{@link ByteBuf#readableBytes()}, you might want to strip the length
field by specifying <tt>initialBytesToStrip</tt>. In this example, we
specified <tt>2</tt>, that is same with the length of the length field, to
strip the first two bytes.

* BEFORE DECODE (14 bytes)         AFTER DECODE (12 bytes)
* | Length | Actual Content |----->| Actual Content |
* | 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
* +----------------+      +----------------+
* </pre>

* In most cases, the length field represents the length of the message body
* only, as shown in the previous examples. However, in some protocols, the
* length field represents the length of the whole message, including the
* message header. In such a case, we specify a non-zero
* <tt>lengthAdjustment</tt>. Because the length value in this example message
* is always greater than the body length by <tt>2</tt>, we specify <tt>-2</tt>
* as <tt>lengthAdjustment</tt> for compensation.

* BEFORE DECODE (14 bytes)         AFTER DECODE (14 bytes)
* | Length | Actual Content |----->| Actual Content |
* | 0x000C | "HELLO, WORLD" |      | "HELLO, WORLD" |
* +----------------+      +----------------+
* </pre>

* <h3>3 bytes length field at the end of 5 bytes header, do not strip header</h3>
* The following message is a simple variation of the first example. An extra
* header value is prepended to the message. <tt>lengthAdjustment</tt> is zero
* again because the decoder always takes the length of the prepended data into
* account during frame length calculation.
* <pre>
* <b>lengthFieldOffset</b>   = <b>2</b> (= the length of Header 1)
* <b>lengthFieldLength</b>   = <b>3</b>
* lengthAdjustment    = 0
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* | Header 1 |  Length  | Actual Content |----->| Header 1 |  Length  | Actual Content |
* | 0xCAFE  | 0x00000C | "HELLO, WORLD" |      | 0xCAFE  | 0x00000C | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>
*
* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>
*
* This is an advanced example that shows the case where there is an extra
* header between the length field and the message body. You have to specify a
* positive <tt>lengthAdjustment</tt> so that the decoder counts the extra
* header into the frame length calculation.
* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* |  Length  | Header 1 | Actual Content |----->|  Length  | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE  | "HELLO, WORLD" |      | 0x00000C | 0xCAFE  | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>
*
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header,
* strip the first header field and the length field</h3>
*
* This is a combination of all the examples above. There are the prepended
* header before the length field and the extra header after the length field.
* The prepended header affects the <tt>lengthFieldOffset</tt> and the extra
* header affects the <tt>lengthAdjustment</tt>. We also specified a non-zero
* <tt>initialBytesToStrip</tt> to strip the length field and the prepended
* header from the frame. If you don't want to strip the prepended header, you
* could specify <tt>0</tt> for <tt>initialBytesToSkip</tt>.
* <pre>
* lengthFieldOffset   = 1 (= the length of HDR1)
* lengthFieldLength  = 2
* <b>lengthAdjustment</b>   = <b>1</b> (= the length of HDR2)
* <b>initialBytesToStrip</b> = <b>3</b> (= the length of HDR1 + LEN)

* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +------+--------+------+----------------+      +------+----------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +------+--------+------+----------------+      +------+----------------+
* </pre>

* 2 bytes length field at offset 1 in the middle of 4 bytes header,
*  strip the first header field and the length field, the length field
*  represents the length of the whole message

* Let's give another twist to the previous example. The only difference from
*  the previous example is that the length field represents the length of the
*  whole message instead of the message body, just like the third example.
*  We have to count the length of HDR1 and Length into <tt>lengthAdjustment</tt>.
*  Please note that we don't need to take the length of HDR2 into account
*  because the length field already includes the whole header length.

* BEFORE DECODE (16 bytes) AFTER DECODE (13 bytes)
* +------+--------+------+----------------+      +------+----------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x0010 | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +------+--------+------+----------------+      +------+----------------+
* </pre>

* @see LengthFieldPrepender

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* https://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

No license file was found, but licenses were detected in source scan.

/
*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/Crc32c.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/JdkZlibDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/xml/XmlFrameDecoder.java

Found in path(s):
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/LzfDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/Lz4FrameEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/FastLz.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/FastLzFrameEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/FastLzFrameDecoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1-
  jar/io/netty/handler/codec/compression/Lz4Constants.java
Open Source Used In DNAC Platform Arica 40002

* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/AsciiHeadersEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/compression/Bzip2MoveToFrontTable.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/compression/LzmaFrameEncoder.java
* /opt/cola/permits/1145449520_1616155795.37/0/netty-codec-4-1-59-final-sources-1.jar/io/netty/handler/codec/compression/MessageAggregationException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

1.2865 scala-reflect 2.13.5

1.2865.1 Available under license:

Scala
Copyright (c) 2002-2020 EPFL
Copyright (c) 2011-2020 Lightbend, Inc.

Scala includes software developed at
LAMP/EPFL (https://lamp.epfl.ch/) and
Lightbend, Inc. (https://www.lightbend.com/).

Licensed under the Apache License, Version 2.0 (the "License").
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This software includes projects with other licenses -- see `doc/LICENSE.md`.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2866 gson 2.2.4

1.2867 hamcrest 1.3

1.2867.1 Available under license :

```html
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<html>
<head>
<title>Common Public License - v 1.0</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" vlink="#800000">

<p align="center"><b>Common Public License - v 1.0</b></p><p>
<font size="3"></font></p><font size="2">THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.</font></p><font size="2">
<p><font size="2">1. DEFINITIONS</font></p><font size="2">
<p><font size="2">a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and</font></p><font size="2">
<p><font size="2">b) in the case of each subsequent Contributor:</font></p><font size="2">
<p><font size="2">i) changes to the Program, and</font></p><font size="2">
<p><font size="2">ii) additions to the Program;</font></p><font size="2">
```
where such changes and/or additions to the Program originate from and are distributed by
that particular Contributor. A Contribution 'originates' from a Contributor if it was
added to the Program by such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include additions to the Program which: (i) are separate modules of software
distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of
the Program.

Contributor means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily
infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance
with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all
Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide,
royalty-free copyright license to
reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the
Contribution of such Contributor, if any, and such derivative works, in source code and object code
form.

b) Subject to the terms of this Agreement, each Contributor
hereby grants Recipient a non-exclusive, worldwide,
royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the
time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the
patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.</UL>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2">d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement. </UL>

<UL><FONT SIZE="2"></FONT></UL>

<P><B>3. REQUIREMENTS</B></P>

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

<UL><FONT SIZE="2">a) it complies with the terms and conditions of this Agreement; and</FONT></UL>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2">b) its license agreement:</FONT></UL>

<UL><FONT SIZE="2">i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose; </FONT>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2">ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits; </FONT></UL>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2">iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and</FONT></UL>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2">iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.</FONT></UL>

<UL><FONT SIZE="2"></FONT></UL>

<UL><FONT SIZE="2" COLOR="#000FF"> </FONT><FONT SIZE="2" COLOR="#FF0000"><UL><FONT SIZE="2" COLOR="#000FF"></FONT></UL></FONT></UL>
When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program, and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, etc.
programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

BSD License
Copyright (c) 2000-2003, jMock.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2000-2003, jMock.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of jMock nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>EasyMock License</title>
<link rel="stylesheet" href="easymock.css" />
</head>
<body><div class="bodywidth">

<h2>EasyMock License (MIT License)</h2>

Copyright (c) 2001-2006 OFFIS, Tammo Freese.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:
</p>
<p>The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.
</p>
<p>THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
</p>
</div>
</body>
</html>

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2868 nssocket 0.6.0 0.6.0

1.2868.1 Available under license:

Copyright (C) 2011 Charlie Robbins, Paolo Fragomeni, & the Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2869 jetty-util-ajax 9.4.41.v20210516

1.2869.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (“AGREEMENT”). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT’S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

“Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor’s behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

“Contributor” means any person or entity that distributes the Program.

“Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

“Program” means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

   a) it complies with the terms and conditions of this Agreement; and
   b) its license agreement:
      i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
      ii) effectively excludes on behalf of all Contributors all liability for
damages, including direct, indirect, special, incidental and
consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are
offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable
manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained
within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if
any, in a manner that reasonably allows subsequent Recipients to identify the
originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with
respect to end users, business partners and the like. While this license is
intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner
which does not create potential liability for other Contributors. Therefore,
if a Contributor includes the Program in a commercial product offering, such
Contributor ("Commercial Contributor") hereby agrees to defend and indemnify
every other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits and
other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such Commercial
Contributor in connection with its distribution of the Program in a commercial
product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property
infringement. In order to qualify, an Indemnified Contributor must:
a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement
negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product
offering, Product X. That Contributor is then a Commercial Contributor. If
that Commercial Contributor then makes performance claims, or offers
warranties related to Product X, those performance claims and warranties are
such Commercial Contributor's responsibility alone. Under this section, the
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

==============================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
==============================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both
* The Apache 2.0 License
   http://www.apache.org/licenses/LICENSE-2.0.html

   and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat.jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------

MortBay

The following artifacts are CDDL + GPLv2 with classpath exception.
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.2870 wide-align 1.1.3

1.2870.1 Available under license :

Copyright (c) 2015, Rebecca Turner <me@re-becca.org>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2871 camelcase 5.3.1 5.3.1

1.2871.1 Available under license :

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2872 concat-stream 2.0.0 2.0.0

1.2872.1 Available under license :

The MIT License

Copyright (c) 2013 Max Ogden

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2873 readable-stream 1.1.14 1.1.14

1.2873.1 Available under license :

Copyright Joyent, Inc. and other Node contributors. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2874 kerb-simplekdc 1.0.0

1.2874.1 Available under license:

Kerb Simple Kdc
Copyright 2014-2017 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf.
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2875 tough-cookie 2.3.4 2.3.4

1.2875.1 Available under license :

Copyright (c) 2015, Salesforce.com, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the
following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of Salesforce.com nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

===

The following exceptions apply:

===

`public_suffix_list.dat` was obtained from
<https://publicsuffix.org/list/public_suffix_list.dat> via
<http://publicsuffix.org>. The license for this file is MPL/2.0. The header of that file reads as follows:

// This Source Code Form is subject to the terms of the Mozilla Public
// License, v. 2.0. If a copy of the MPL was not distributed with this
// file, You can obtain one at http://mozilla.org/MPL/2.0/.

1.2876 jersey-guice 1.19

1.2877 ws 6.2.2
1.2877.1 Available under license :

(The MIT License)

Copyright (c) 2017-2019 Kamil Myliwiec <http://kamilmyliwiec.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2878 mime 2.6.0

1.2878.1 Available under license:

The MIT License (MIT)

Copyright (c) 2017 Ronomon

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2879 nan 2.10.0 2.10.0

1.2879.1 Available under license:

The MIT License (MIT)

====================================

Copyright (c) 2018 NAN contributors

====================================
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2880 har-validator 2.1.3 2.1.3

1.2880.1 Available under license:

Copyright (c) 2015, Ahmad Nassri <ahmad@ahmadnassri.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2881 fs-writefile-promise 1.0.3 1.0.3

1.2881.1 Available under license:

Copyright (c) 2015, Ahmad Nassri <ahmad@ahmadnassri.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2882 fs-readfile-promise 2.0.1 2.0.1

1.2882.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014 - 2015 Shinnosuke Watanabe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2883 expand-tilde 2.0.2 2.0.2

1.2883.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015-2016, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.


1.2884.1 Available under license:
The MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2885 escape-string-regexp 1.0.5 1.0.5

1.2885.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2886 bean-validation-api 2.0.1.Final
1.2886.1 Available under license:
Bean Validation API

License: Apache License, Version 2.0
See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.

1.2887 co 4.6.0 4.6.0
1.2887.1 Available under license:
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk &lt;tj@vision-media.ca&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
1.2889 freemarker 2.3.28

1.2889.1 Available under license:
Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file
distributed with this work for additional information regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

====================================================================
The source code contains the following binaries, which were created at the Apache FreeMarker project, and hence are covered by the same license as the other source files of it:

src/main/misc/overloadedNumberRules/prices.ods
src/manual/en_US/docgen-originals/figures/overview.odg
src/manual/en_US/docgen-originals/figures/model2sketch_with_alpha.png
src/manual/en_US/docgen-originals/figures/tree_with_alpha.png
src/manual/en_US/favicon.png
src/manual/en_US/figures/model2sketch.png
src/manual/en_US/figures/overview.png
src/manual/en_US/figures/tree.png
src/manual/en_US/logo.png
src/manual/zh_CN/favicon.png
src/manual/zh_CN/figures/model2sketch.png
src/manual/zh_CN/figures/overview.png
src/manual/zh_CN/figures/tree.png
src/manual/zh_CN/logo.png

Apache FreeMarker
Copyright 2015-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

------------------------------------------------------------------------

The Apache FreeMarker documentation contains the following binaries,
which were created at the Apache FreeMarker project, and hence are
covered by the same license as the other source files of it:

documentation/_html/docgen-resources/img/patterned-bg.png
documentation/_html/docgen-resources/img/xxe.png
documentation/_html/favicon.png
documentation/_html/figures/model2sketch.png
documentation/_html/figures/overview.png
documentation/_html/figures/tree.png
documentation/_html/logo.png
The documentation includes a selection of icons from various icon sets (fonts), stored together inside these files, which were generated with https://icomoon.io/app/:

documentation/_html/docgen-resources/fonts/icomoon.eot
documentation/_html/docgen-resources/fonts/icomoon.svg
documentation/_html/docgen-resources/fonts/icomoon.ttf
documentation/_html/docgen-resources/fonts/icomoon.woff

The name, license, and attribution of each icon sets (fonts) used follows:

- The documentation includes icons from Entypo pictograms, version 2.0, by Daniel Bruce (http://www.entypo.com/, http://www.danielbruce.se/), licensed under Creative Commons Attribution-ShareAlike 3.0 (CC BY-SA 3.0) (http://creativecommons.org/licenses/by-sa/3.0/legalcode) and under SIL Open Font License 1.1 (http://scripts.sil.org/OFL).

- The documentation includes icons from Font Awesome by Dave Gandy (http://fontawesome.io), licensed under SIL Open Font License 1.1 (http://scripts.sil.org/OFL).

- The documentation includes icons from Material Design icons by Google (http://google.github.io/material-design-icons/), licensed under Creative Common Attribution 4.0 International License (CC-BY 4.0) (https://creativecommons.org/licenses/by/4.0/).

=========================================================================
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2890 ansi-regex 2.1.1 2.1.1
1.2890.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2891 body-parser 1.18.3 1.18.3
1.2891.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
1.2892 reflections 0.9.10
1.2892.1 Available under license:
DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004

Copyright (C) 2004 Sam Hocevar <sam@hocevar.net>

Everyone is permitted to copy and distribute verbatim or modified copies of this license document, and changing it is allowed as long as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

1.2893 slash 1.0.0
1.2893.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2894 should.js 13.2.3 13.2.3
1.2894.1 Available under license:
Copyright (c) 2010-2013 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2013-2017 Denis Bardadym <bardadymchik@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2895 babel-plugin-transform-regenerator
6.26.0

1.2895.1 Available under license:
Copyright (C) 2012-2014 by various contributors (see AUTHORS)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
The << Software identified by reference to the Ecma Standard* ("Software") >> is protected by copyright and is being made available under the "BSD License", included below. This Software may be subject to third party rights (rights from parties other than Ecma International), including patent rights, and no licenses under such third party rights are granted under this license even if the third party concerned is a member of Ecma International. SEE THE ECMA CODE OF CONDUCT IN PATENT MATTERS AVAILABLE AT http://www.ecma-international.org/memento/codeofconduct.htm FOR INFORMATION REGARDING THE LICENSING OF PATENT CLAIMS THAT ARE REQUIRED TO IMPLEMENT ECMA INTERNATIONAL STANDARDS*.

Copyright (C) 2012-2013 Ecma International
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the authors nor Ecma International may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE ECMA INTERNATIONAL "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ECMA INTERNATIONAL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* Ecma International Standards hereafter means Ecma International Standards as well as Ecma Technical Reports
Copyright (c) 2014-2015 Jason Quense <jason@quense.me>
Original work by respective rule authors; copywrites noted in files.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (c) jQuery Foundation, Inc. and Contributors, All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015 Jed Watson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2896 once 1.4.0 1.4.0
1.2896.1 Available under license :

   The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
1.2897 sprintf-js 1.0.3 1.0.3

1.2897.1 Available under license:

Copyright (c) 2007-2014, Alexandru Marasteanu <hello [at] alexei (dot) ro>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of this software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2898 better-assert 1.0.2 1.0.2

1.2898.1 Available under license:

Copyright (c) 2013 Nathan Rajlich &lt;nathan@tootallnate.net&gt;

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2899 mute-stream 0.0.7 0.0.7
1.2899.1 Available under license:
The ISC License

Copyright (c) Isaac Z. Schlueter and Contributors

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2900 bytes 3.0.0 3.0.0
1.2900.1 Available under license:
(The MIT License)

Copyright (c) 2012-2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Jed Watson <jed.watson@me.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2901 @exodus/schemasafe 1.0.0-rc.3 1.0.0-rc.3
1.2901.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Mathias Buus
Copyright (c) 2020 Exodus Movement

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2902 object.assign 4.1.2
1.2902.1 Available under license:
The MIT License (MIT)

Copyright (c) 2014 Jordan Harband
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2903 jackson-datatype-jdk8 2.10.5

1.2903.1 Available under license:

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1595301005506
Build-Jdk: 1.8.0_162
Built-By: tatu
Bundle-Description: Add-on module for Jackson (http://jackson.codehaus.org) to support JDK 8 data types.
Bundle-DocURL: https://github.com/FasterXML/jackson-modules-java8/jackson-datatype-jdk8
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: Jackson datatype: jdk8
Bundle-SymbolicName: com.fasterxml.jackson.datatype.jackson-datatype-jdk8
Bundle-Vendor: FasterXML
Bundle-Version: 2.10.5
Created-By: Apache Maven Bundle Plugin
Export-Package: com.fasterxml.jackson.datatype.jackson-datatype-jdk8;version="2.10.5";uses="com.fasterxml.jackson.core,jackson.databind.deser.std,com.fasterxml.jackson.databind.type,com.fasterxml.jackson.databind.util"
1.2904 google-discovery-to-swagger 2.1.0

2.1.0

1.2904.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015 APIs-guru

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

Open Source Used In DNAC Platform Arica 14073
1.2905 librsvg 2.40.13-3

1.2905.1 Available under license:

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to
permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Library General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>  

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

  Gnomovision version 69, Copyright (C) year  name of author  
  Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
  This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

**1.2906 pupa 2.1.1**

**1.2906.1 Available under license :**

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**1.2907 is-property 1.0.2**

**1.2907.1 Available under license :**

No license file was found, but licenses were detected in source scan.

(c) 2013 Mikola Lysenko. MIT License
1.2908 @babel/plugin-proposal-numeric-separator 7.16.0

1.2908.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2909 process-nextick-args 1.0.7 1.0.7

1.2909.1 Available under license:

# Copyright (c) 2015 Calvin Metcalf

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

**THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.**

1.2910 hbase-logging 2.4.2

1.2910.1 Available under license:

Apache HBase - Logging
Copyright 2007-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made, 
use, offer to sell, sell, import, and otherwise transfer the Work, 
where such license applies only to those patent claims licensable 
by such Contributor that are necessarily infringed by their 
Contribution(s) alone or by combination of their Contribution(s) 
with the Work to which such Contribution(s) was submitted. If You 
institute patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Work 
or a Contribution incorporated within the Work constitutes direct 
or contributory patent infringement, then any patent licenses 
granted to You under this License for that Work shall terminate 
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the 
Work or Derivative Works thereof in any medium, with or without 
modifications, and in Source or Object form, provided that You 
meet the following conditions:

(a) You must give any other recipients of the Work or 
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices 
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works 
   that You distribute, all copyright, patent, trademark, and 
   attribution notices from the Source form of the Work, 
   excluding those notices that do not pertain to any part of 
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its 
   distribution, then any Derivative Works that You distribute must 
   include a readable copy of the attribution notices contained 
   within such NOTICE file, excluding those notices that do not 
   pertain to any part of the Derivative Works, in at least one 
   of the following places: within a NOTICE text file distributed 
   as part of the Derivative Works; within the Source form or 
   documentation, if provided along with the Derivative Works; or, 
   within a display generated by the Derivative Works, if and 
   wherever such third-party notices normally appear. The contents 
   of the NOTICE file are for informational purposes only and 
   do not modify the License. You may add Your own attribution 
   notices within Derivative Works that You distribute, alongside 
   or as an addendum to the NOTICE text from the Work, provided 
   that such additional attribution notices cannot be construed 
   as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2911 nopt 1.0.10 1.0.10

1.2911.1 Available under license:

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2912 @babel/plugin-transform-unicode-escapes 7.16.0

1.2912.1 Available under license:

MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2913 deep-is 0.1.3 0.1.3

1.2913.1 Available under license:

Copyright (c) 2012, 2013 Thorsten Lorenz <thlorenz@gmx.de>
Copyright (c) 2012 James Halliday <mail@substack.net>
Copyright (c) 2009 Thomas Robinson <280north.com>
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2914 drill-format-httpd 1.19.0

1.2914.1 Available under license:

Drill : Contrib : Format : Httpd/Nginx Access Log
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2915 strip-bom 4.0.0

1.2915.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2916 oauth-sign 0.8.2 0.8.2
1.2916.1 Available under license :
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1
through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are
under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of
fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source
code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form,
including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as
indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
1.2917 define-property 1.0.0 1.0.0

1.2917.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2918 archy 1.0.0

1.2918.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2919 vector 1.19.0

1.2919.1 Available under license:

<#--
Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

-->
/
*
* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Drill : Exec : Vectors
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing. Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS.
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
1.2920 netty-transport-native-unix-common
4.1.59.Final
1.2920.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources.jar/io/netty/channel/unix/DomainSocketReadMode.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources.jar/io/netty/channel/unix/DatagramSocketAddress.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources.jar/io/netty/channel/unix/NativeInetAddress.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources.jar/io/netty/channel/unix/Socket.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources.jar/io/netty/channel/unix/Errors.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/io/netty/channel/unix/UnixChannel.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_socket.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_errors.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_filedescriptor.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_socket.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_fil descriptor.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_errors.h
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/netty_unix_jni.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-jar/io/netty/channel/unix/UnixChannelUtil.java
No license file was found, but licenses were detected in source scan.

/
* Copyright 2018 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/netty_unix_buffer.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/netty_unix_buffer.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/io/netty/channel/unix/Buffer.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*  https://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/io/netty/channel/unix/LimitsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
  jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/netty_unix_util.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/netty_unix_limits.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/netty_unix_util.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/netty_unix_limits.c
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/io/netty/channel/unix/PeerCredentials.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/io/netty/channel/unix/Limits.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/io/netty/channel/unix/SocketWritableByteChannel.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/netty_unix.h
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native unix-common-4-1-59-final-sources-jar/netty_unix.c

No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE
~ 2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/io/netty/channel/unix/UnixChannelOption.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/io/netty/channel/unix/Unix.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1138680848_1614312229.62/0/netty-transport-native-unix-common-4-1-59-final-sources-
jar/io/netty/channel/unix/IovArray.java

1.2921 protostuff-json 1.7.1

1.2921.1 Available under license :
No license file was found, but licenses were detected in source scan.

//Licensed under the Apache License, Version 2.0 (the "License");
//you may not use this file except in compliance with the License.
//You may obtain a copy of the License at
//http://www.apache.org/licenses/LICENSE-2.0
//distributed under the License is distributed on an "AS IS" BASIS,
1.2922 summernote 0.8.11 0.8.11

1.2922.1 Available under license:
The MIT License (MIT)

Copyright (c) 2015~ Summernote Team (https://github.com/orgs/summernote/people)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2923 jackson-dataformat-yaml 2.9.8

1.2923.1 Available under license:
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.
## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

### 1.2924 hsqldb-jdbc 1.8.0.10

1.2924.1 Available under license :

```java
/*
 * For work developed by the HSQL Development Group:
 * *
 * Copyright (c) 2001-2010, The HSQL Development Group
 * All rights reserved.
 * *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 * *
 * Redistributions of source code must retain the above copyright notice, this
 * list of conditions and the following disclaimer.
 * *
 * Redistributions in binary form must reproduce the above copyright notice,
 * this list of conditions and the following disclaimer in the documentation
 * and/or other materials provided with the distribution.
 * *
 * Neither the name of the HSQL Development Group nor the names of its
 * contributors may be used to endorse or promote products derived from this
 * software without specific prior written permission.
 * *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
```
* ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
* For work originally developed by the Hypersonic SQL Group:
*
* Copyright (c) 1995-2000, The Hypersonic SQL Group.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
*
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*
* Neither the name of the Hypersonic SQL Group nor the names of its
* contributors may be used to endorse or promote products derived from this
* software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE HYPERSONIC SQL GROUP,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
* This software consists of voluntary contributions made by many individuals
* on behalf of the Hypersonic SQL Group.
*/
/* Copyright (c) 2001-2010, The HSQL Development Group
* All rights reserved.
*/
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
* 
* Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
* 
* Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
* 
* Neither the name of the HSQL Development Group nor the names of its
* contributors may be used to endorse or promote products derived from this
* software without specific prior written permission.
* 
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG,
* OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

1.2925 has-symbols 1.0.2

1.2925.1 Available under license:
MIT License

Copyright (c) 2016 Jordan Harband

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

Open Source Used In DNAC Platform Arica 14121
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2926 mkdirp-promise 1.1.0 1.1.0
1.2926.1 Available under license :
Copyright (c) 2015, Ahmad Nassri <ahmad@ahmadnassri.com>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2927 jsreport-pptx 0.7.0 0.7.0
1.2927.1 Available under license :
The MIT License (MIT)

Copyright (c) 2016 jsreport

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2928 string-width 2.1.1 2.1.1

1.2928.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2929 drill-storage-cassandra 1.19.0

1.2929.1 Available under license:

Drill : Contrib : Storage : Cassandra
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, \WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2930 lodash 4.17.19 4.17.19

1.2930.1 Available under license:

Copyright OpenJS Foundation and other contributors <https://openjsf.org/>

Based on Underscore.js, copyright Jeremy Ashkenas, DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

1.2931 curl 7.51.0
1.2931.1 Available under license:
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2020, Daniel Stenberg, <daniel@haxx.se>, and many contributors, see the THANKS file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

1.2932 javax-el-api 2.2.4
1.2932.1 Available under license:

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License <bottom> Copyright (c) 1999-2011 Oracle and/or its affiliates. All Rights Reserved. Use is subject to license terms. </bottom> Found license 'General Public License 2.0' in '# Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. # The contents of this file are subject to the terms of either the GNU # General Public License Version 2 only ("GPL") or the Common Development # and Distribution License("CDDL") (collectively, the "License"). You # may not use this file except in compliance with the License. You can # Oracle designates this particular file as subject to the "Classpath" # exception as provided by Oracle in the GPL Version 2 section of the License # Licensed under the Apache License, Version 2.0 (the "License"); # you may not use this file except in compliance with the
License. # You may obtain a copy of the License at # http://www.apache.org/licenses/LICENSE-2.0 # distributed under the License is distributed on an "AS IS" BASIS.
Found license 'General Public License 2.0' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'General Public License 2.0' in 'Copyright (c) 1997-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'General Public License 2.0' in 'Copyright (c) 1999-2011 Oracle and/or its affiliates. All Rights Reserved. Use is subject to license terms. <bottom>'
Found license 'General Public License 2.0' in '* Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * http://www.apache.org/licenses/LICENSE-2.0 * distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'General Public License 2.0' in '* Copyright (c) 2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'General Public License 2.0' in '* Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'General Public License 2.0' in 'Copyright (c) 1997-2011 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2010 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 distributed under the License is distributed on an "AS IS" BASIS.'
1.2933 make-dir 2.1.0

1.2933.1 Available under license:

MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2934 joda-time 2.9.9

1.2934.1 Available under license:

=============================================================================
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
=============================================================================
This product includes software developed by
Joda.org (http://www.joda.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed...
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
y any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2935 commander 2.20.3 2.20.3
1.2935.1 Available under license:
(The MIT License)

Copyright (c) 2011 TJ Holowaychuk <tj@vision-media.ca>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2936 parquet-encoding 1.8.1
1.2936.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This project includes code from Daniel Lemire's JavaFastPFOR project in this binary artifact. The "Lemire" bit packing classes produced by parquet-generator are derived from the JavaFastPFOR project.

Copyright: 2013 Daniel Lemire
Home page: http://lemire.me/en/
Project page: https://github.com/lemire/JavaFastPFOR
License: Apache License Version 2.0 http://www.apache.org/licenses/LICENSE-2.0

Apache Parquet Encodings
Copyright 2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.2937 debug 2.3.3 2.3.3
1.2937.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2938 content-disposition 0.5.3 0.5.3

1.2938.1 Available under license :
(The MIT License)

Copyright (c) 2014-2017 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2939 get-stream 4.1.0 4.1.0

1.2939.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2940 p-try 2.2.0 2.2.0
1.2940.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2941 @babel/plugin-proposal-private-property-in-object 7.16.0
1.2941.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY. WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2942 zstd 1.4.9

1.2942.1 Available under license:
BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to
  endorse or promote products derived from this software without specific
  prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.2943 amcharts3-animate 1.1.2 1.1.2

1.2943.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

Open Source Used In DNAC Platform Arica 14153
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2944 should-format 3.0.3 3.0.3

1.2944.1 Available under license :

The MIT License (MIT)

Copyright (c) 2014

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2945 debug 2.2.0 2.2.0
1.2945.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2946 commons-compress 1.14
1.2946.1 Available under license:

Apache Commons Compress
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The files in the package org.apache.commons.compress.archivers.sevenz
were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/),
which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICEx text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.2947 okio 2.7.0

1.2947.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/-Util.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/-Utf8.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/-Platform.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/ByteString.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/ByteString.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-Platform.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ForwardingTimeout.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/SegmentedByteString.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
*/

* Copyright (C) 2017 Square, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2018 Square, Inc.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/PeekSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-DeprecatedUtf8.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Throttler.kt
No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2016 Square, Inc.
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* Copyright (C) 2019 Square, Inc.

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/okio/Pipe.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Options.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/HashingSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/HashingSource.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/RealBufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/RealBufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/SegmentedByteString.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Sink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/BufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Buffer.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Source.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/Buffer.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/RealBufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Timeout.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/BufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/RealBufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Okio.kt

No license file was found, but licenses were detected in source scan.

/
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio-/Base64.kt
HEAR YE, HEAR YE!

All of the java software and associated documentation about the DOM in this repository are distributed under the license from the W3C, which is provided herein.

LICENSE.dom-software.txt covers all software from the W3C including the following items in the xml-commons project:

    xml-commons/java/external/src/org/w3c
    and all subdirectories

(Note: SAC (The Simple API for CSS) has been published under an older version of the W3C license. The original license file is LICENSE.sac.html.)

LICENSE.dom-documentation.txt covers all documentation from the W3C including the following items in the xml-commons project:

    xml-commons/java/external/xdocs/dom
    and all subdirectories

The actual DOM Java Language Binding classes in xml-commons came from:
http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.html

The specification of DOM Level 3’s various parts is at:
http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/
http://www.w3.org/TR/2004/REC-DOM-Level-3-LS-20040407/
http://www.w3.org/TR/2004/NOTE-DOM-Level-3-XPath-20040226/

The specification of DOM Level 2’s various parts is at:
http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Traversal-Range-20001113/
http://www.w3.org/TR/2000/REC-DOM-Level-2-Views-20001113/

The specification of DOM Level 1’s various parts is at:
Links to all available W3C DOM Java Bindings can be found at:
  http://www.w3.org/DOM/DOMTR

The actual classes of The Simple API for CSS (SAC) came from:
  http://www.w3.org/Style/CSS/SAC/
  http://www.w3.org/2002/06/sacjava-1.3.zip

The actual DOM Java Language Binding classes for SMIL came from:
  http://dev.w3.org/cvsweb/java/classes/org/w3c/dom/smil/
  (both ElementTimeControl.java and TimeEvent.java were taken at revision 1.1)

The actual DOM Java Language Binding classes for SVG 1.1 came from:
  http://www.w3.org/TR/SVG11/java.html

=========================================================================
==  NOTICE file corresponding to section 4(d) of the Apache License,     ==
==  Version 2.0, in this case for the Apache Xerces distribution.      ==
=========================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
  - voluntary contributions made by Paul Eng on behalf of the
    Apache Software Foundation that were originally developed at iClick, Inc.,
    software copyright (c) 1999.

BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=========================================================================  
==  NOTICE file corresponding to section 4(d) of the Apache License,   ==  
==  Version 2.0, in this case for the Apache Xalan serializer          ==  
==  distribution.                                                      ==  
=========================================================================  

This product includes software developed by IBM Corporation (http://www.ibm.com) and The Apache Software Foundation (http://www.apache.org/).

Portions of this software was originally based on the following:
- software copyright (c) 1999-2002, Lotus Development Corporation.,
- software copyright (c) 2001-2002, Sun Microsystems.,


This license came from: http://www.w3.org/Consortium/Legal/copyright-software-20021231

W3C SOFTWARE NOTICE AND LICENSE
http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the
redistributed or derivative work.

2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.

3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) $

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
Commons Logging

Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by 'an unknown organization'
- Unnamed - avalon-framework:avalon-framework:jar:4.1.3
- Unnamed - log4j:log4j:jar:1.2.12
- Unnamed - logkit:logkit:jar:1.0.1

Apache Santuario - XML Security for Java
Copyright 1999-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

It was originally based on software copyright (c) 2001, Institute for

The development of this software was partly funded by the European
Commission in the <WebSig> project in the ISIS Programme.


HEAR YE, HEAR YE!

All of the .java software and associated documentation about
SAX in this repository are distributed freely in the
public domain.

LICENSE.sax.txt covers all software and documentation from the
megginson.com including the following in the xml-commons project:

xml-commons/java/external/src/org/xml/sax
   and all subdirectories
xml-commons/java/external/xdocs/sax
   and all subdirectories

The actual SAX classes in xml-commons came from:
http://www.megginson.com/Software/index.html
Apache XML Commons XML APIs
Copyright 2006 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xalan distribution. ==
=====================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of this software was originally based on the following:
- software copyright (c) 1999-2002, Lotus Development Corporation.,
- software copyright (c) 2001-2002, Sun Microsystems.,
- voluntary contributions made by Ovidiu Predescu <ovidiu@cup.hp.com> on
  behalf of the Apache Software Foundation that was originally developed
  at Hewlett Packard Company.

The binary distribution package (ie. jars, samples and documentation) of
this product includes software developed by the following:
- The Apache Software Foundation
- Xerces Java - see xercesImpl.README.txt
- JAXP 1.3 APIs - see xml-apis.README.txt
- Bytecode Engineering Library - see BCEL.README.txt
- Regular Expression - see regexp.README.txt

- Scott Hudson, Frank Flannery, C. Scott Ananian
- CUP Parser Generator runtime (javacup\runtime) - see runtime.README.txt

The source distribution package (ie. all source and tools required to build Xalan Java) of this product includes software developed by the following:

- The Apache Software Foundation
  - Xerces Java - see xercesImpl.README.txt
  - JAXP 1.3 APIs - see xml-apis.README.txt
  - Bytecode Engineering Library - see BCEL.README.txt
  - Regular Expression - see regexp.README.txt
  - Ant - see ant.README.txt
  - Stylebook doc tool - see stylebook-1.0-b3_xalan-2.README.txt

- Elliot Joel Berk and C. Scott Ananian
  - Lexical Analyzer Generator (JLex) - see JLex.README.txt

========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
========================================================================

Apache Xerces Java
Copyright 1999-2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.

This license came from: http://www.w3.org/Consortium/Legal/copyright-documents-20021231

W3C DOCUMENT LICENSE
Public documents on the W3C site are provided by the copyright holders under the following license. By using and/or copying this document, or the W3C document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [$date-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all
times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle $Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) $

=========================================================================== NOTICE file corresponding to section 4(d) of the Apache License, == Version 2.0, in this case for the Apache xml-commons xml-apis == distribution. ==
===========================================================================

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000 World Wide Web Consortium, http://www.w3.org


This license came from: http://www.megginson.com/SAX/copying.html
However please note future versions of SAX may be covered under http://saxproject.org/?selected=pdf

This page is now out of date -- see the new SAX site at
http://www.saxproject.org/ for more up-to-date releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and
SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2949 is-number 2.1.0

1.2949.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-2015, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2950 lodash-es 4.17.10 4.17.10
1.2950.1 Available under license:

Copyright JS Foundation and other contributors <https://js.foundation/>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <http://underscorejs.org/>

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/lodash/lodash

The following license applies to all parts of this software except as
documented below:

=====

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

Copyright and related rights for sample code are waived via CC0. Sample
code is defined as all source code displayed within the prose of the
documentation.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

=====

Files located in the node_modules and vendor directories are externally
maintained libraries used by this software which have their own
licenses; we recommend you read them, as their terms may differ from the
terms above.
1.2951 aop-alliance 1.0
1.2951.1 Available under license:
   Public Domain

1.2952 debug 4.1.1 4.1.1
1.2952.1 Available under license:
   (The MIT License)
   Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>

   Permission is hereby granted, free of charge, to any person obtaining a copy of this software
   and associated documentation files (the 'Software'), to deal in the Software without restriction,
   including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
   and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
   subject to the following conditions:

   The above copyright notice and this permission notice shall be included in all copies or substantial
   portions of the Software.

   THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
   INCLUDING BUT NOT
   LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
   AND NONINFRINGEMENT.
   IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
   DAMAGES OR OTHER LIABILITY,
   WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
   CONNECTION WITH THE
   SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2953 resolve 1.20.0
1.2953.1 Available under license:
   MIT License
   Copyright (c) 2012 James Halliday

   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2954 busboy 0.2.14 0.2.14

1.2954.1 Available under license:
Copyright Brian White. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.2955 setimmediate 1.0.5 1.0.5

1.2955.1 Available under license:
Copyright (c) 2012 Barnesandnoble.com, llc, Donavon West, and Domenic Denicolà

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2956 array-each 1.0.1 1.0.1

1.2956.1 Available under license:

The MIT License (MIT)

Copyright (c) 2015, 2017, Jon Schlinkert

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2957 fill-range 7.0.1

1.2957.1 Available under license:

The MIT License (MIT)

Copyright (c) 2014-present, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2958 accessors-smart 1.2

1.2958.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2011 JSON-SMART authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-jar/net/minidev/asm/BeansAccess.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-jar/net/minidev/asm/DynamicClassLoader.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-jar/net/minidev/asm/Accessor.java
* /opt/cola/permits/1177937262_1625568439.63/0/accessors-smart-1-2-sources-5-jar/net/minidev/asm/ASMUtil.java
1.2959 amcharts3-export 1.4.73 1.4.73

1.2959.1 Available under license:

Copyright (C) 2012-2015  SheetJS

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Except where noted, this license applies to any and all software programs and associated documentation files created
by the Original Author and distributed with the Software:

'jszip.js' is a modified version of JSZip, Copyright (c) Stuart Knightley, David Duponchel, Franz Buchinger,
Antonio Afonso. JSZip is dual licensed and is used according to the terms of the MIT License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2960 orc-core 1.5.6
1.2960.1 Available under license :

ORC Core
Copyright 2013-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2961 cookie-parser 1.4.4 1.4.4
1.2961.1 Available under license :
(The MIT License)

Copyright (c) 2014 TJ Holowaychuk <tj@vision-media.ca>
Copyright (c) 2015 Douglas Christopher Wilson <doug@somethingdoug.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2962 split-string 3.1.0 3.1.0
1.2962.1 Available under license :
The MIT License (MIT)

Copyright (c) 2015-2017, Jon Schlinkert.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2963 npm-install-package 2.1.0 2.1.0

1.2963.1 Available under license :

The MIT License (MIT)

Copyright (c) 2015 Yoshua Wuyts

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2964 jackson 2.2.3

1.2964.1 Available under license :

FasterXML, LLC
Software Grant and Corporate Contributor License Agreement ("Agreement")
http://fasterxml.com/clia
(v1)
Thank you for your interest in FasterXML, LLC ("FasterXML"). In order to clarify the intellectual property license granted with Contributions from any person or entity, FasterXML must have a Contributor License Agreement (CLA) on file that has been signed by each Contributor, indicating agreement to the license terms below. This license is for your protection as a Contributor as well as the protection of FasterXML and its users; it does not change your rights to use your own Contributions for any other purpose.

This version of the Agreement allows an entity (the "Corporation") to submit Contributions to the FasterXML, to authorize Contributions submitted by its designated employees to FasterXML, and to grant copyright and patent licenses thereto.

If you have not already done so, please complete and sign, then scan and email a pdf file of this Agreement to clas@fasterxml.com. If necessary, send an original signed Agreement to FasterXML, LLC, 600 N 36th Ave, Suite 409, Seattle, WA 98103.

Please read this document carefully before signing and keep a copy for your records.

Corporation name: ________________________________________________

Corporation address: ________________________________________________

..............................................................................................

..............................................................................................

Point of Contact: ________________________________________________

E-Mail: ________________________________________________

Telephone: _____________________ Fax: _____________________

You accept and agree to the following terms and conditions for Your present and future Contributions submitted to FasterXML. Except for the license granted herein to FasterXML and recipients of software distributed by FasterXML, You reserve all right, title, and interest in and to Your Contributions.

1. Definitions.

"You" (or "Your") shall mean the copyright owner or legal entity authorized by the copyright owner that is making this Agreement
with FasterXML. For legal entities, the entity making a
Contribution and all other entities that control, are controlled
by, or are under common control with that entity are considered
to be a single Contributor. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of
the outstanding shares, or (iii) beneficial ownership of such
entity.

"Contribution" shall mean the code, documentation or other
original works of authorship expressly identified in Schedule B,
as well as any original work of authorship, including any
modifications or additions to an existing work, that is
intentionally submitted by You to FasterXML for inclusion in, or
documentation of, any of the products owned or managed by
FasterXML (the "Work"). For the purposes of this definition,
"submitted" means any form of electronic, verbal, or written
communication sent to FasterXML or its representatives,
including but not limited to communication on electronic mailing
lists, source code control systems, and issue tracking systems
that are managed by, or on behalf of, FasterXML for the purpose
of discussing and improving the Work, but excluding
communication that is conspicuously marked or otherwise
designated in writing by You as "Not a Contribution."

2. Grant of Copyright License. Subject to the terms and conditions
of this Agreement, You hereby grant to FasterXML and to
recipients of software distributed by FasterXML a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare derivative works of,
publicly display, publicly perform, sublicense, and distribute
Your Contributions and such derivative works.

3. Grant of Patent License. Subject to the terms and conditions of
this Agreement, You hereby grant to FasterXML and to recipients
of software distributed by FasterXML a perpetual, worldwide,
non-exclusive, no-charge, royalty-free, irrevocable (except as
stated in this section) patent license to make, have made, use,
offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by You that are necessarily infringed by Your Contribution(s)
alone or by combination of Your Contribution(s) with the Work to
which such Contribution(s) were submitted. If any entity institutes
patent litigation against You or any other entity (including a
cross-claim or counterclaim in a lawsuit) alleging that your
Contribution, or the Work to which you have contributed, constitutes
direct or contributory patent infringement, then any patent licenses
granted to that entity under this Agreement for that Contribution or Work shall terminate as of the date such litigation is filed.

4. You represent that You are legally entitled to grant the above license. You represent further that each employee of the Corporation designated on Schedule A below (or in a subsequent written modification to that Schedule) is authorized to submit Contributions on behalf of the Corporation.

5. You represent that each of Your Contributions is Your original creation (see section 7 for submissions on behalf of others).

6. You are not expected to provide support for Your Contributions, except to the extent You desire to provide support. You may provide support for free, for a fee, or not at all. Unless required by applicable law or agreed to in writing, You provide Your Contributions on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

7. Should You wish to submit work that is not Your original creation, You may submit it to FasterXML separately from any Contribution, identifying the complete details of its source and of any license or other restriction (including, but not limited to, related patents, trademarks, and license agreements) of which you are personally aware, and conspicuously marking the work as "Submitted on behalf of a third-party: [named here]."

8. It is your responsibility to notify FasterXML when any change is required to the list of designated employees authorized to submit Contributions on behalf of the Corporation, or to the Corporation's Point of Contact with FasterXML.

Please sign: __________________________ Date: ______________
Title: __________________________
Corporation: __________________________

Schedule A

[Initial list of designated employees. NB: authorization is not tied to particular Contributions.]
1.2965 send 0.17.1 0.17.1

1.2965.1 Available under license:
(The MIT License)

Copyright (c) 2012 TJ Holowaychuk
Copyright (c) 2014-2016 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2966 qdox 1.12.1

1.2967 drill-common 1.19.0

1.2967.1 Available under license:

Drill : Common
Copyright 2021 The Apache Software Foundation
This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2968 jline-terminal-jansi 3.12.1
1.2968.1 Available under license:

No license file was found, but licenses were detected in source scan.

2017, the original author or authors.

This software is distributable under the BSD license. See the terms of the
BSD license in the documentation provided with this software.

https://opensource.org/licenses/BSD

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-jar/META-INF/maven/org.jline/jline-terminal-jansi/pom.xml

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2017, the original author or authors.
* 
* This software is distributable under the BSD license. See the terms of the
* BSD license in the documentation provided with this software.
* 
* https://opensource.org/licenses/BSD-3-Clause
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-jar/org/jline/terminal/impl/jansi/JansiNativePty.java
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-jar/org/jline/terminal/impl/jansi/solaris/SolarisNativePty.java
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-jar/org/jline/terminal/impl/jansi/osx/OsXNativePty.java
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009-2018 the original author(s).
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-
jar/org/jline/terminal/impl/jansi/win/Windows AnsiWriter.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2002-2019, the original author or authors.
 *
 * This software is distributable under the BSD license. See the terms of the
 * BSD license in the documentation provided with this software.
 *
 * https://opensource.org/licenses/BSD-3-Clause
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1057261525_1592255684.35/0/jline-terminal-jansi-3-12-1-sources-
jar/org/jline/terminal/impl/jansi/win/JansiWinSysTerminal.java

1.2969 jackson 2.11.4
1.2969.1 Available under license:

=========================================================================
== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0, ==
== in this case for the Apache Camel distribution. ==
=========================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Please read the different LICENSE files present in the licenses directory of
this distribution.

Camel :: Jackson
Copyright 2007-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2970 detect-file 1.0.0 1.0.0

1.2970.1 Available under license:

The MIT License (MIT)

Copyright (c) 2016-2017, Brian Woodward.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

1.2971 hikaricp-java7 2.4.13

1.2972 proxy-addr 1.0.10 1.0.10

1.2972.1 Available under license:

(The MIT License)

Copyright (c) 2014 Douglas Christopher Wilson

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2973 cassandra-driver-core 3.2.0

1.2974 is-absolute-url 2.1.0 2.1.0
1.2974.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2975 wordwrap 1.0.0 1.0.0

1.2975.1 Available under license:

This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2976 popperjs 1.15.0 1.15.0

1.2976.1 Available under license:

No license file was found, but licenses were detected in source scan.

Code and documentation copyright 2016 **Federico Zivolo**. Code released under the [MIT license](LICENSE.md). Docs released under Creative Commons.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/README.md
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper.js

/**!
* @fileOverview Kickass library to create and place poppers near their reference elements.
* @version 1.15.0
* @license
* Copyright (c) 2016 Federico Zivolo and contributors
* 
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
* copies of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
* 
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
* 
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper.js
No license file was found, but licenses were detected in source scan.

/*
Copyright (C) Federico Zivolo 2019
Distributed under the MIT License (license terms are at http://opensource.org/licenses/MIT).
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/popper.min.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/esm/popper.min.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper-utils.min.js
* /opt/ws_local/PERMITS_SQL/1025666388_1594212234.43/0/popper-js-1-15-0-tgz/package/dist/umd/popper.min.js

1.2977 @babel/plugin-proposal-export-namespace-from 7.16.0
1.2977.1 Available under license:
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2978 is-obj 2.0.0

1.2978.1 Available under license :
MIT License

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2979 object-assign 2.1.1 2.1.1

1.2979.1 Available under license :
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1.2980 tslib 1.14.1 1.14.1
1.2980.1 Available under license:
/

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) Microsoft Corporation.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

1.2981 unicode-canonical-property-names-ecmascript 2.0.0
1.2981.1 Available under license:
Copyright Mathias Bynens <https://mathiasbynens.be/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2982 lowercase-keys 1.0.1
1.2982.1 Available under license:
The MIT License (MIT)

Copyright (c) Sindre Sorhus <sindresorhus@gmail.com> (sindresorhus.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2983 oas-validator 5.0.5
1.2983.1 Available under license:
BSD 3-Clause License

Copyright (c) 2016, Mermade Software
All rights reserved.

Redistribution and use in source and binary forms, with or without
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2984 engine.io-client 3.4.4 3.4.4
1.2984.1 Available under license :
(The MIT License)

Copyright (c) 2014-2015 Automattic <dev@cloudup.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
1.2985 ignite-indexing 2.11.0
1.2985.1 Available under license:

ignite-indexing
Copyright 2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

// List of ignite-indexing module's dependencies provided as a part of this distribution.

For H2 Database Engine (http://www.h2database.com) com.h2database:h2:jar:1.4.197
This product bundles H2 Database Engine which is available under the following:
MPL 2.0 or EPL 1.0. For details, see http://h2database.com/html/license.html.

For ignite-core (http://ignite.apache.org) org.apache.ignite:ignite-core:jar:2.11.0
This product bundles ignite-core which is available under the following:
The Apache Software License, Version 2.0. For details, see http://www.apache.org/licenses/LICENSE-2.0.txt.

org.apache.lucene:lucene-analyzers-common:jar:7.4.0
This product bundles Lucene Common Analyzers which is available under the following:
Apache 2. For details, see http://www.apache.org/licenses/LICENSE-2.0.txt.

This product bundles Lucene Core which is available under the following:
Apache 2. For details, see http://www.apache.org/licenses/LICENSE-2.0.txt.

This product bundles Lucene QueryParsers which is available under the following:
Apache 2. For details, see http://www.apache.org/licenses/LICENSE-2.0.txt.

This product bundles Apache Commons Codec which is available under the following:
Apache License, Version 2.0. For details, see https://www.apache.org/licenses/LICENSE-2.0.txt.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2986 commons-daemon 1.0.13

1.2986.1 Available under license:

Apache Commons Daemon
Copyright 1999-2013 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and...
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2987 underscore 1.9.2 1.9.2

1.2987.1 Available under license :

Copyright (c) 2009-2018 Jeremy Ashkenas, DocumentCloud and Investigative
Reporters & Editors

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restrictions, including without limitation the rights to use, 
copy, modify, merge, publish, distribute, sublicense, and/or sell 
copies of the Software, and to permit persons to whom the 
Software is furnished to do so, subject to the following 
conditions:

The above copyright notice and this permission notice shall be 
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, 
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES 
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND 
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT 
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, 
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING 
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR 
OTHER DEALINGS IN THE SOFTWARE.

1.2988 hive-contrib 3.1.2
1.2988.1 Available under license:

Hive Contrib
Copyright 2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, 
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by 
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all 
other entities that control, are controlled by, or are under common 
control with that entity. For the purposes of this definition, 
"control" means (i) the power, direct or indirect, to cause the 
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.2989 regenerator-runtime 0.11.1 0.11.1
1.2989.1 Available under license :
MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.2990 resolve 1.9.0

1.2990.1 Available under license:
This software is released under the MIT license:

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2991 cairo 1.14.6

1.2991.1 Available under license:
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Cairo is free software.
cairo-trace is released under the terms of the GNU General Public License (GPL) version 3. Please see COPYING-GPL-3 for the precise terms and conditions.

Cairo is free software.

These tests are mainly available under a liberal MIT license to simplify any use of the code for reference purposes. Please check the opening comment of each file for copyright and licensing information.

The test suite also bundles some fonts for use by the test suite. The fonts included, their licenses, and why we use them in the test suite are as follows:

<table>
<thead>
<tr>
<th>Font</th>
<th>License</th>
<th>Distinguishing feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>6x13.pcf</td>
<td>Public Domain</td>
<td>Bitmap font</td>
</tr>
</tbody>
</table>

The test suite also bundles some images for use by the test suite. The images included, their licenses, and why we use them in the test suite are as follows:

<table>
<thead>
<tr>
<th>Image</th>
<th>License</th>
<th>Distinguishing feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>romedalen.jpg</td>
<td>Public Domain</td>
<td>Bitmap image (image/jpeg)</td>
</tr>
<tr>
<td>romedalen.png</td>
<td>Public Domain</td>
<td>Bitmap image (image/png)</td>
</tr>
</tbody>
</table>

The kind contributors of the bundled files are (in alphabetical order):

yvind Kols <pippin@freedesktop.org> Author of the original romedalen shot.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source
form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.
You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as
long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent
"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the
form of a separately written license, or stated as exceptions; 
the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly 
provided under this License. Any attempt otherwise to propagate or 
modify it is void, and will automatically terminate your rights under 
this License (including any patent licenses granted under the third 
paragraph of section 11).

However, if you cease all violation of this License, then your 
license from a particular copyright holder is reinstated (a) 
provisionally, unless and until the copyright holder explicitly and 
finally terminates your license, and (b) permanently, if the copyright 
holder fails to notify you of the violation by some reasonable means 
prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is 
reinstated permanently if the copyright holder notifies you of the 
violation by some reasonable means, this is the first time you have 
received notice of violation of this License (for any work) from that 
copyright holder, and you cure the violation prior to 30 days after 
your receipt of the notice.

Termination of your rights under this section does not terminate the 
licenses of parties who have received copies or rights from you under 
this License. If your rights have been terminated and not permanently 
reinstated, you do not qualify to receive new licenses for the same 
material under section 10.


You are not required to accept this License in order to receive or 
rune a copy of the Program. Ancillary propagation of a covered work 
occurring solely as a consequence of using peer-to-peer transmission 
to receive a copy likewise does not require acceptance. However, 
nothing other than this License grants you permission to propagate or 
modify any covered work. These actions infringe copyright if you do 
not accept this License. Therefore, by modifying or propagating a 
covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically 
receives a license from the original licensors, to run, modify and 
propagate that work, subject to this License. You are not responsible 
for enforcing compliance by third parties with this License.
An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license,
and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published
by the Free Software Foundation.

If the Program specifies that a proxy can decide which future
versions of the GNU General Public License can be used, that proxy's
public statement of acceptance of a version permanently authorizes you
to choose that version for the Program.

Later license versions may give you additional or different
permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a
later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

Cairo is free software.

These tests are mainly available under a liberal MIT license to simplify any use of the code for reference purposes. Please check the opening comment of each file for copyright and licensing information.
Cairo is free software.

Every source file in the implementation of cairo is available to be redistributed and/or modified under the terms of either the GNU Lesser General Public License (LGPL) version 2.1 or the Mozilla Public License (MPL) version 1.1. Some files are available under more liberal terms, but we believe that in all cases, each file may be used under either the LGPL or the MPL.

See the following files in this directory for the precise terms and conditions of either license:
COPYING-LGPL-2.1
COPYING-MPL-1.1

Please see each file in the implementation for copyright and licensing information, (in the opening comment of each file).
Cairo is free software.

These utilities are all free software, please see the licensing conditions in the opening comments of each file.
Cairo is free software.

Every source file in the implementation[*] of cairo is available to be redistributed and/or modified under the terms of either the GNU Lesser General Public License (LGPL) version 2.1 or the Mozilla Public License (MPL) version 1.1. Some files are available under more liberal terms, but we believe that in all cases, each file may be used under either the LGPL or the MPL.

See the following files in this directory for the precise terms and conditions of either license:

COPYING-LGPL-2.1
COPYING-MPL-1.1

Please see each file in the implementation for copyright and licensing information, (in the opening comment of each file).

[*] The implementation of cairo is contained entirely within the "src" directory of the cairo source distribution. There are other components of the cairo source distribution (such as the "test", "util", and "perf") that are auxiliary to the library itself. None of the source code in these directories contributes to a build of the cairo library itself, (libcairo.so or cairo.dll or similar).

These auxiliary components are also free software, but may be under different license terms than cairo itself. For example, most of the test cases in the perf and test directories are made available under an MIT license to simplify any use of this code for reference purposes in using cairo itself. Other files might be available under the GNU General Public License (GPL), for example. Again, please see the COPYING file under each directory and the opening comment of each file for copyright and licensing information.
This is the Debian package of the Cairo multi-platform 2D graphics library

Packaged by Dave Beckett <dajobe@debian.org>

It was downloaded from http://cairographics.org/snapshots/
Copyright 1999 Tom Tromey
Copyright 2002, 2003 University of Southern California, Information Sciences Institute (ISI)
Copyright 2004 Calum Robinson
Copyright 2004 Richard D. Worth
Copyright 2004, 2005 Red Hat, Inc.

Copyright 2004 David Reveman
Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of David Reveman not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. David Reveman makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

DAVID REVEMAN DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL DAVID REVEMAN BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Author: David Reveman <davidr@novell.com>

Cairo is free software.

Every source file in the implementation of cairo is available to be redistributed and/or modified under the terms of either the GNU Lesser General Public License (LGPL) version 2.1 or the Mozilla Public License (MPL) version 1.1. Some files are available under more liberal terms, but we believe that in all cases, each file may be used under either the LGPL or the MPL.

See the following files in this directory for the precise terms and conditions of either license:

COPYING-LGPL-2.1
COPYING-MPL-1.1
Please see each file in the implementation for Copyright and licensing information.

On Debian systems, the complete text of the GNU General Lesser Public License 2.1 can be found in /usr/share/common-licenses/LGPL-2.1

The Mozilla Public License 1.1 (COPYING-MPL-1.1 above) follows:

MOZILLA PUBLIC LICENSE
Version 1.1

1. Definitions.

1.0.1. "Commercial Use” means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor” means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version” means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code” means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism” means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable” means Covered Code in any form other than Source Code.

1.6. "Initial Developer” means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work” means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License” means this document.

1.8.1. "Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or
subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
   A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.
   B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version;
3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
   (a) Third Party Claims.
      If Contributor has knowledge that a license under a third party's
intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the
requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR
ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.


11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such
13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

```
The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is ______________________________________.

The Initial Developer of the Original Code is ____________________.
Portions created by ______________________ are Copyright (C) ______ _________________. All Rights Reserved.

Contributor(s): ______________________________________.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[___] License"), in which case the provisions of [___] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [___] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]"
1.2992 htmlparser2 3.10.1 3.10.1

1.2992.1 Available under license:

Copyright 2010, 2011, Chris Winberry <chris@winberry.net>. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2993 request-local 1.0.5 1.0.5

1.2993.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
“You” (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

“Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

“Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution.”

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following
conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.2994 babel-plugin-proposal-dynamic-import
7.16.0
1.2994.1 Available under license :
MIT License

Copyright (c) 2014-present Sebastian McKenzie and other contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2995 buffer-alloc-unsafe 1.1.0 1.1.0
1.2995.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.2996 jdo-api 3.0.1

1.2997 fined 1.2.0 1.2.0
1.2997.1 Available under license :

The MIT License (MIT)

Copyright (c) 2016, 2017, 2018 Blaine Bublitz <blaine.bublitz@gmail.com> and Eric Schoffstall <yo@contra.io>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.2998 swagger-annotations 1.5.0

1.2998.1 Available under license :

/**
 * Copyright 2015 SmartBear Software
 * <p/>
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * <p/>
 * http://www.apache.org/licenses/LICENSE-2.0
 * <p/>
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

package io.swagger.annotations;

import java.lang.annotation.ElementType;
import java.lang.annotation.Retention;
import java.lang.annotation.RetentionPolicy;
import java.lang.annotation.Target;

/**
 * License metadata available within the info section of a Swagger definition, see
 * https://github.com/swagger-api/swagger-spec/blob/master/versions/2.0.md#licenseObject
 *
 * @since 1.5.0
 */

@Target(ElementType.ANNOTATION_TYPE)
@Retention(RetentionPolicy.RUNTIME)
public @interface License {

/**
 * The name of the license.
 *
 * @return the name of the license
 */
String name();

/**
 * An optional URL for the license.
 */
}
* @return an optional URL for the license.
*
String url() default "";
}

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2022 Cisco Systems, Inc. All rights reserved.